

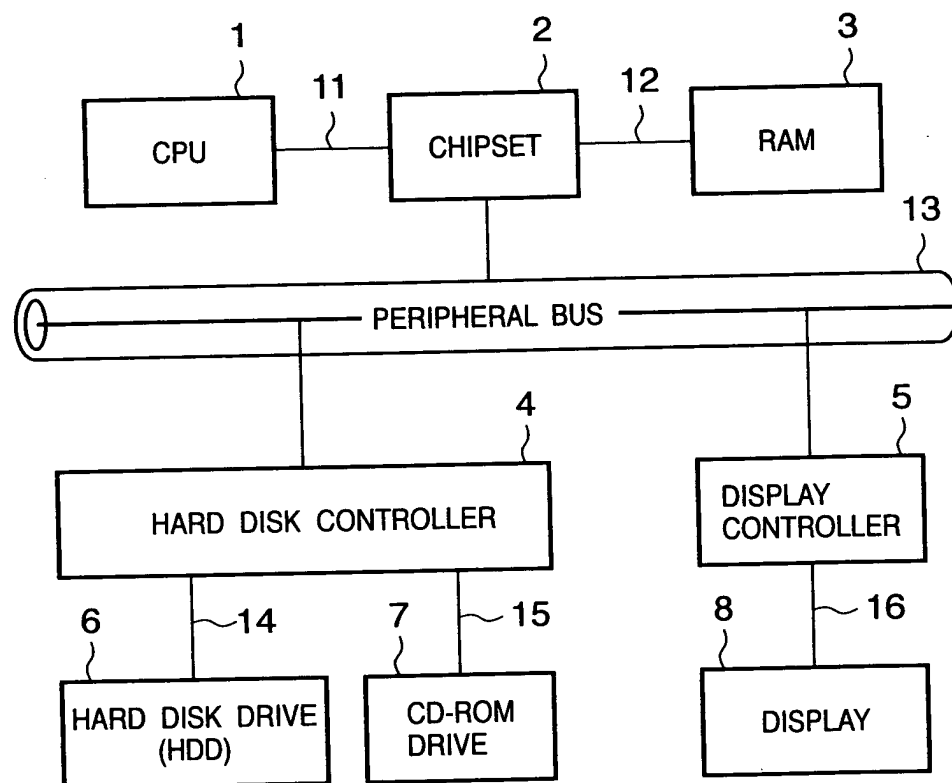
FIG. 1

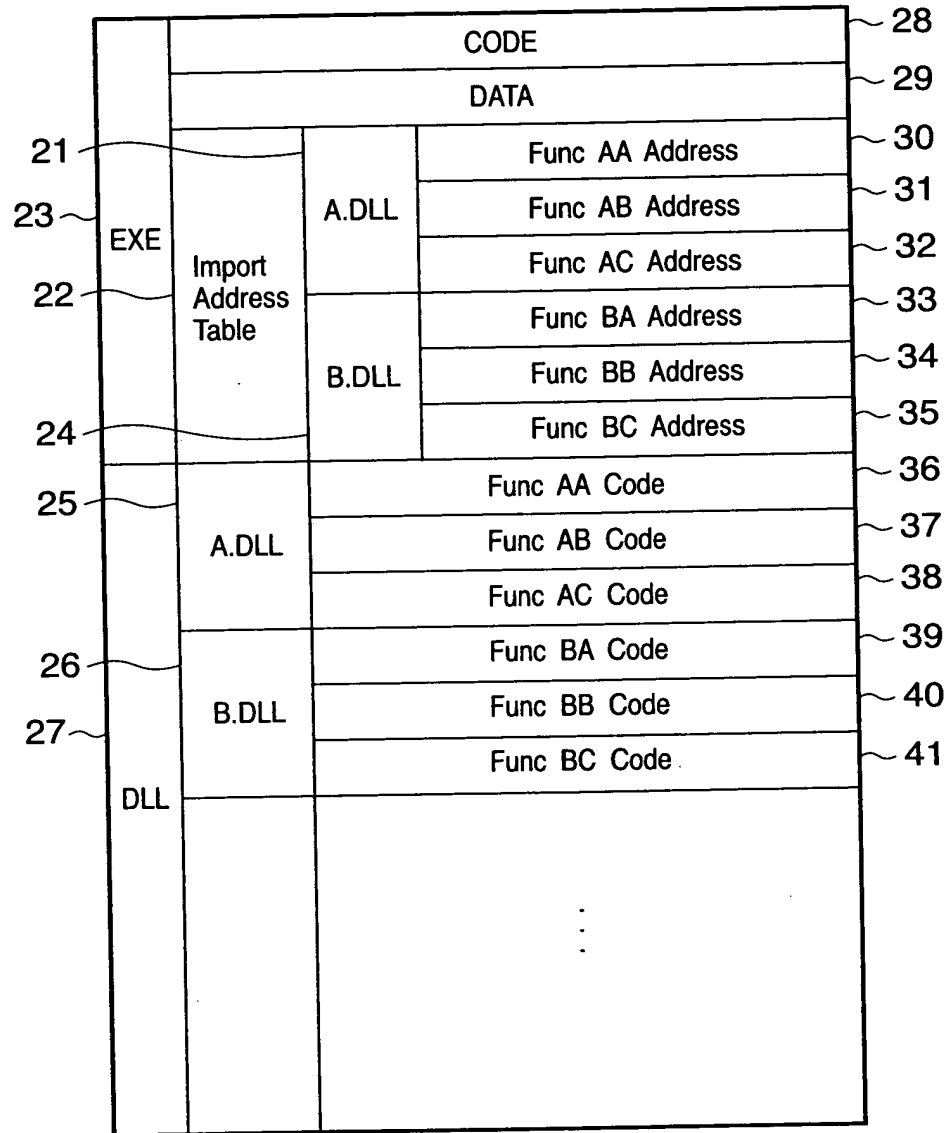
FIG. 2

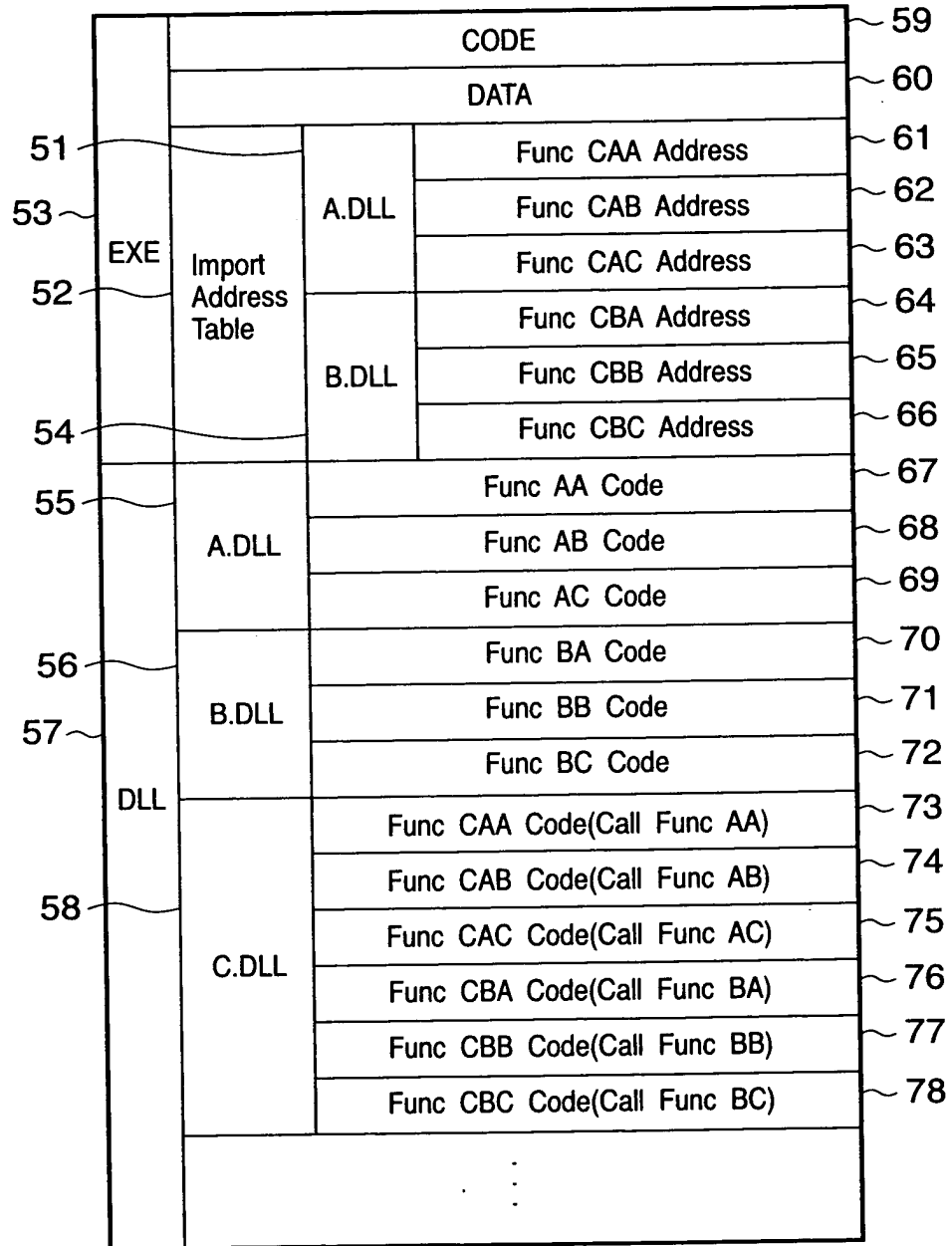
FIG. 3

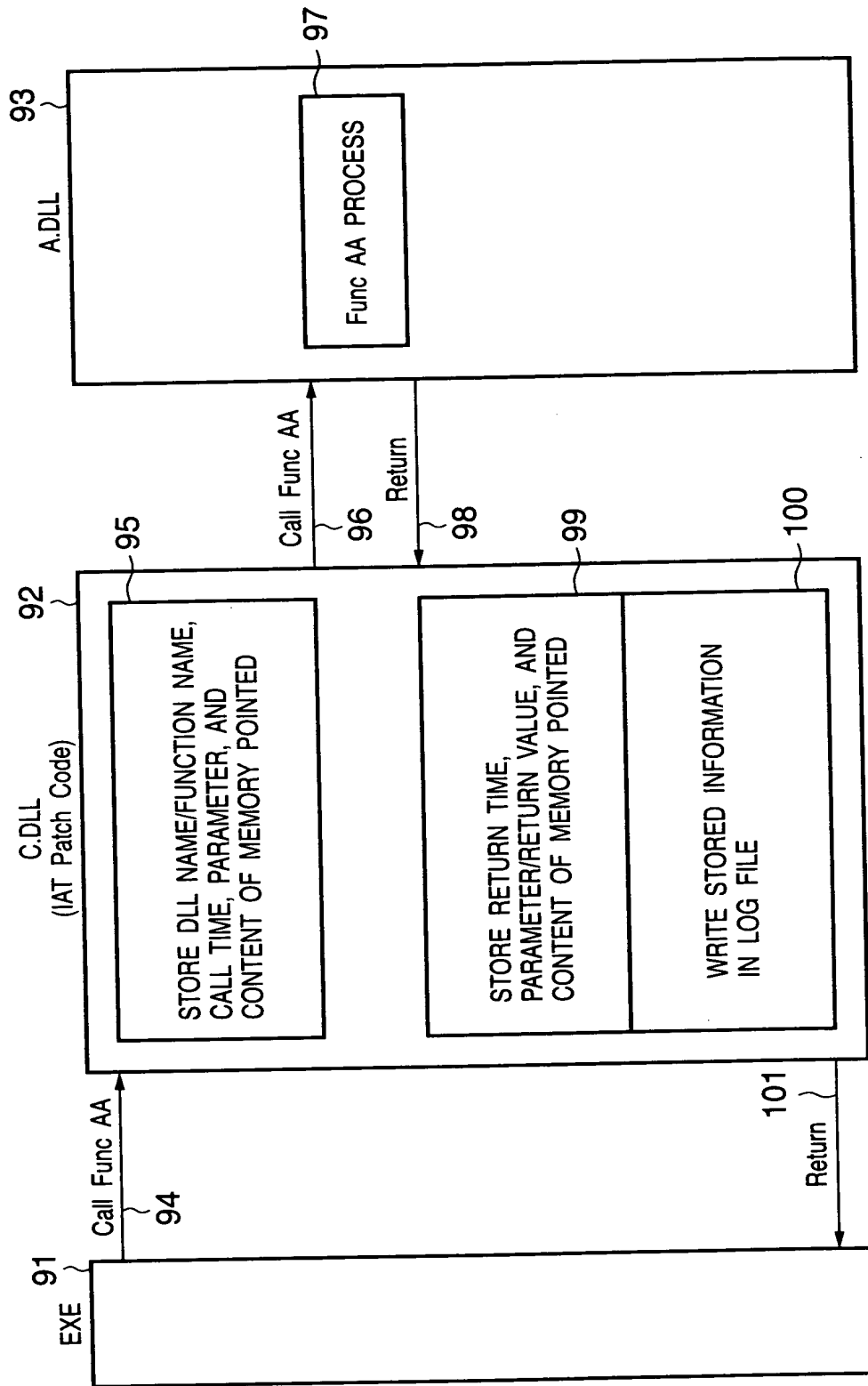
FIG. 4A

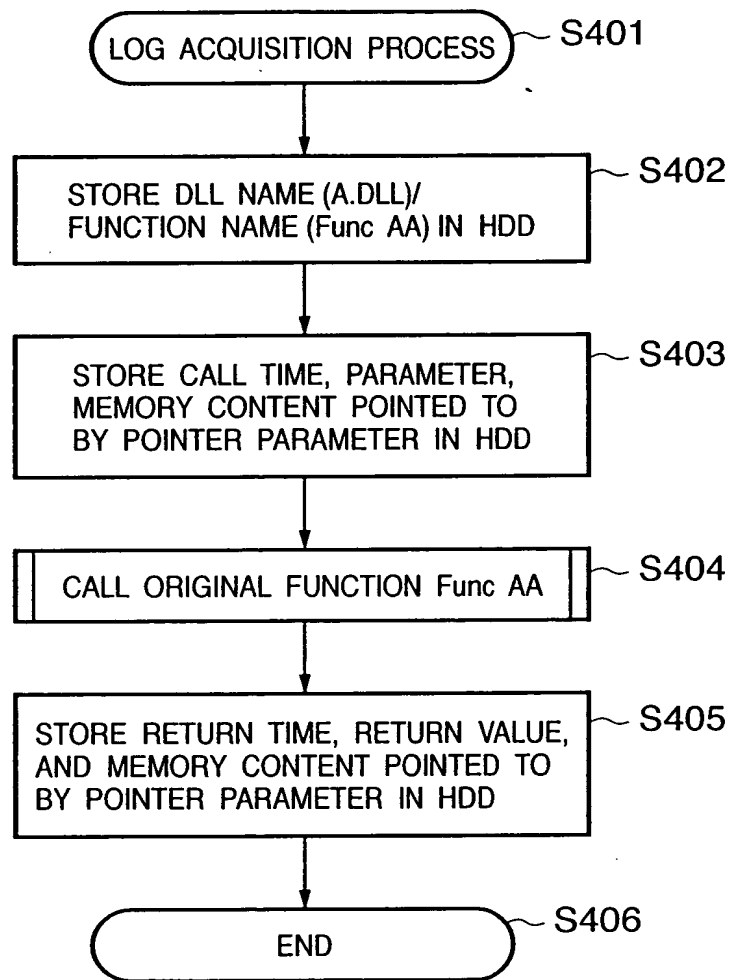
FIG. 4B

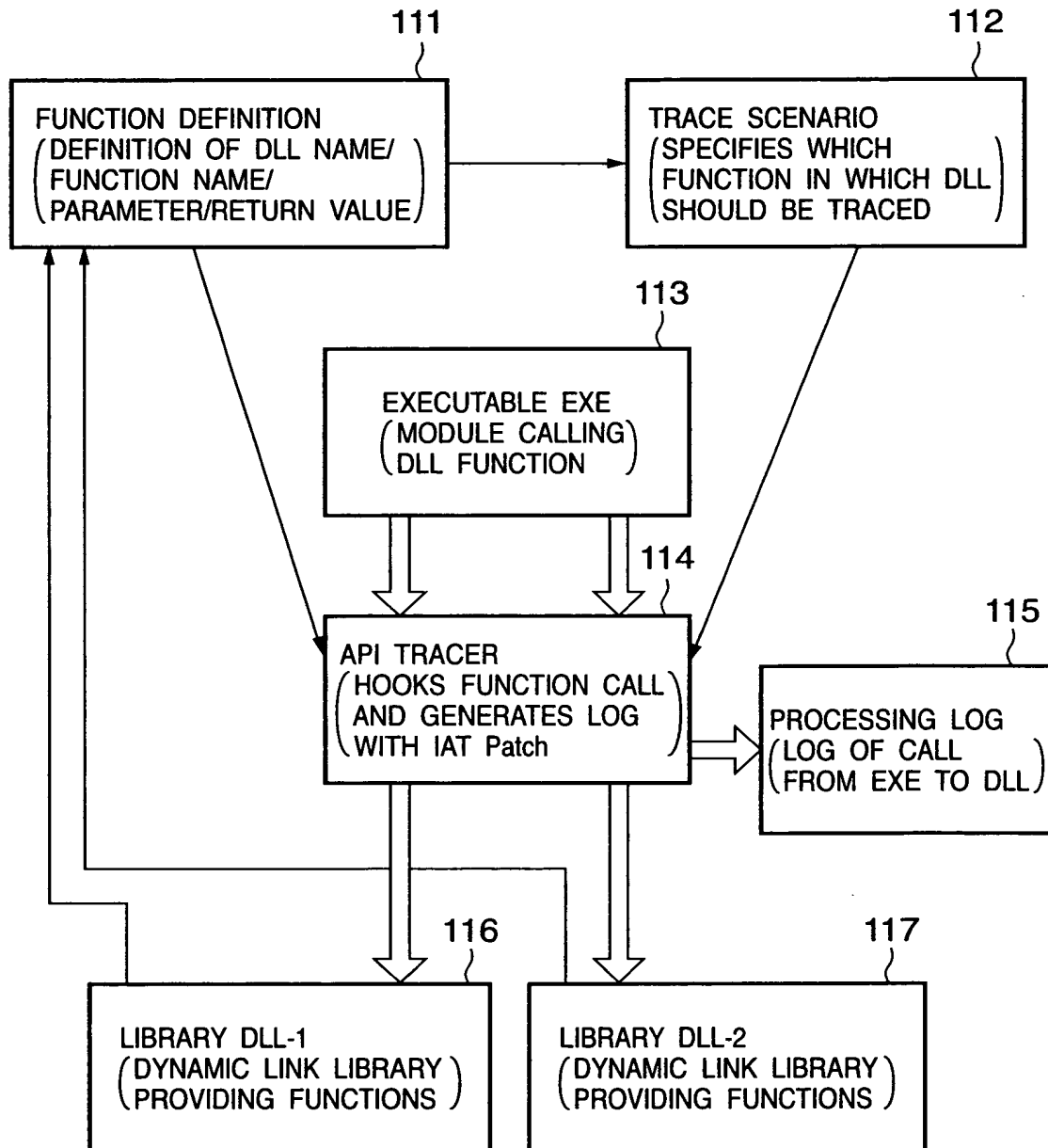
FIG. 5

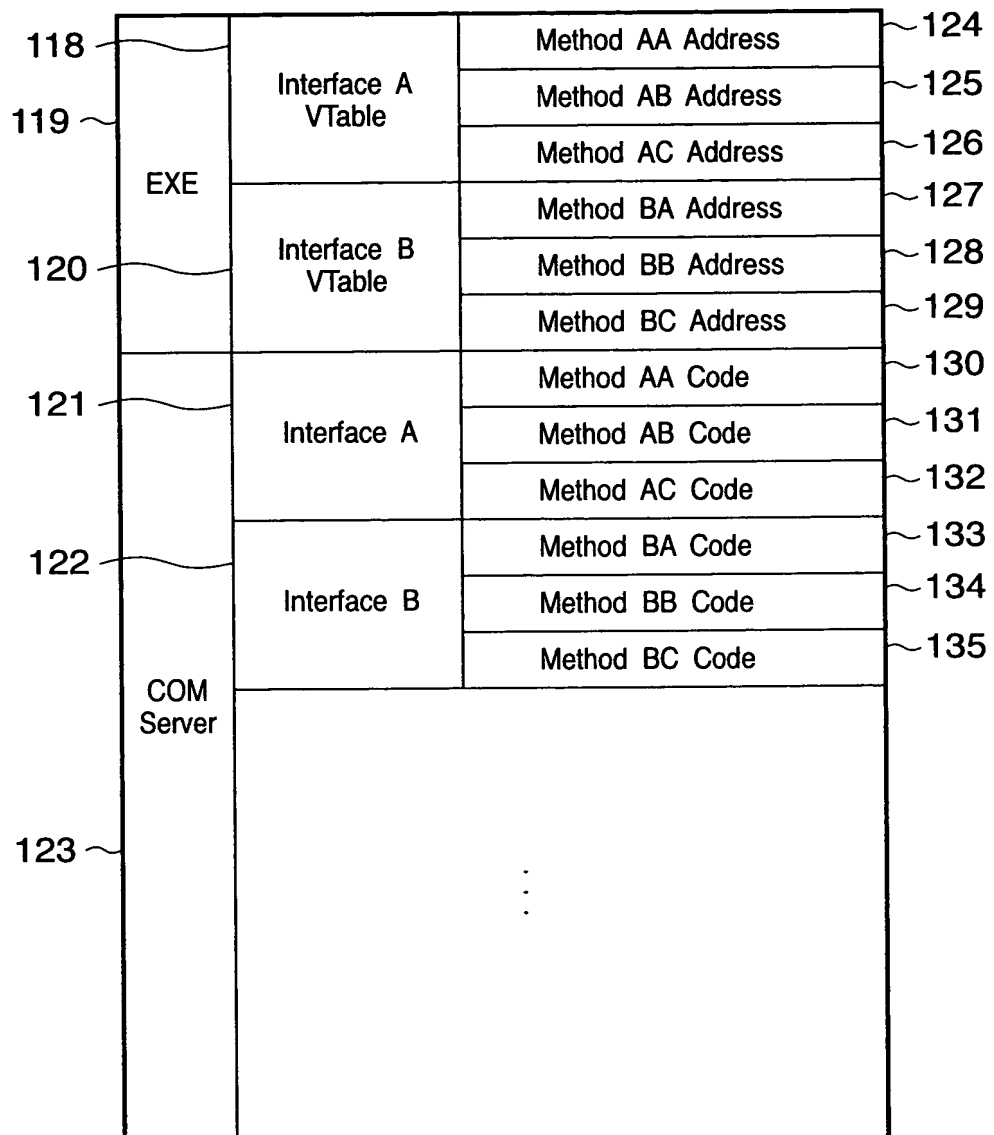
FIG. 6

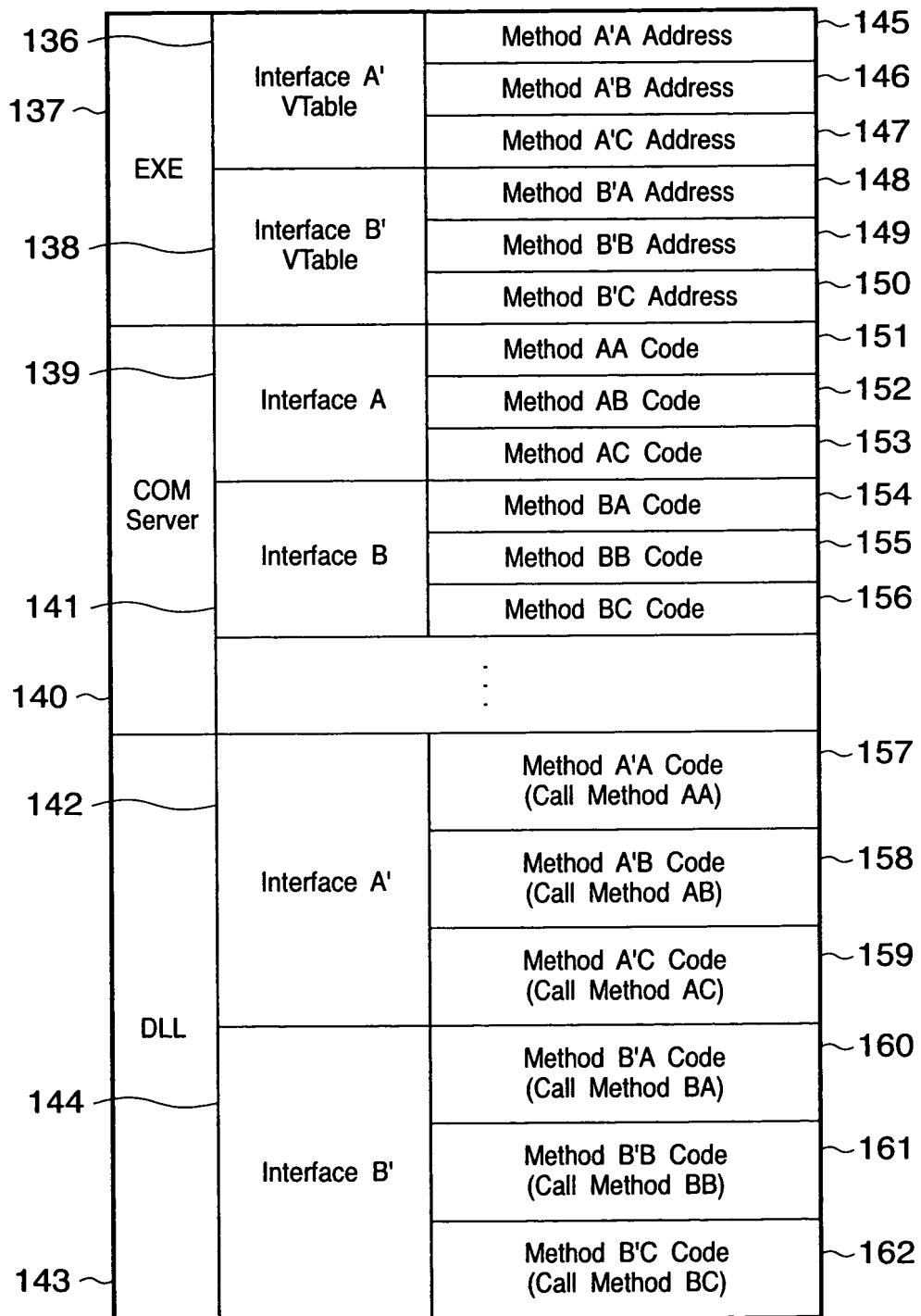
FIG. 7

FIG. 8A

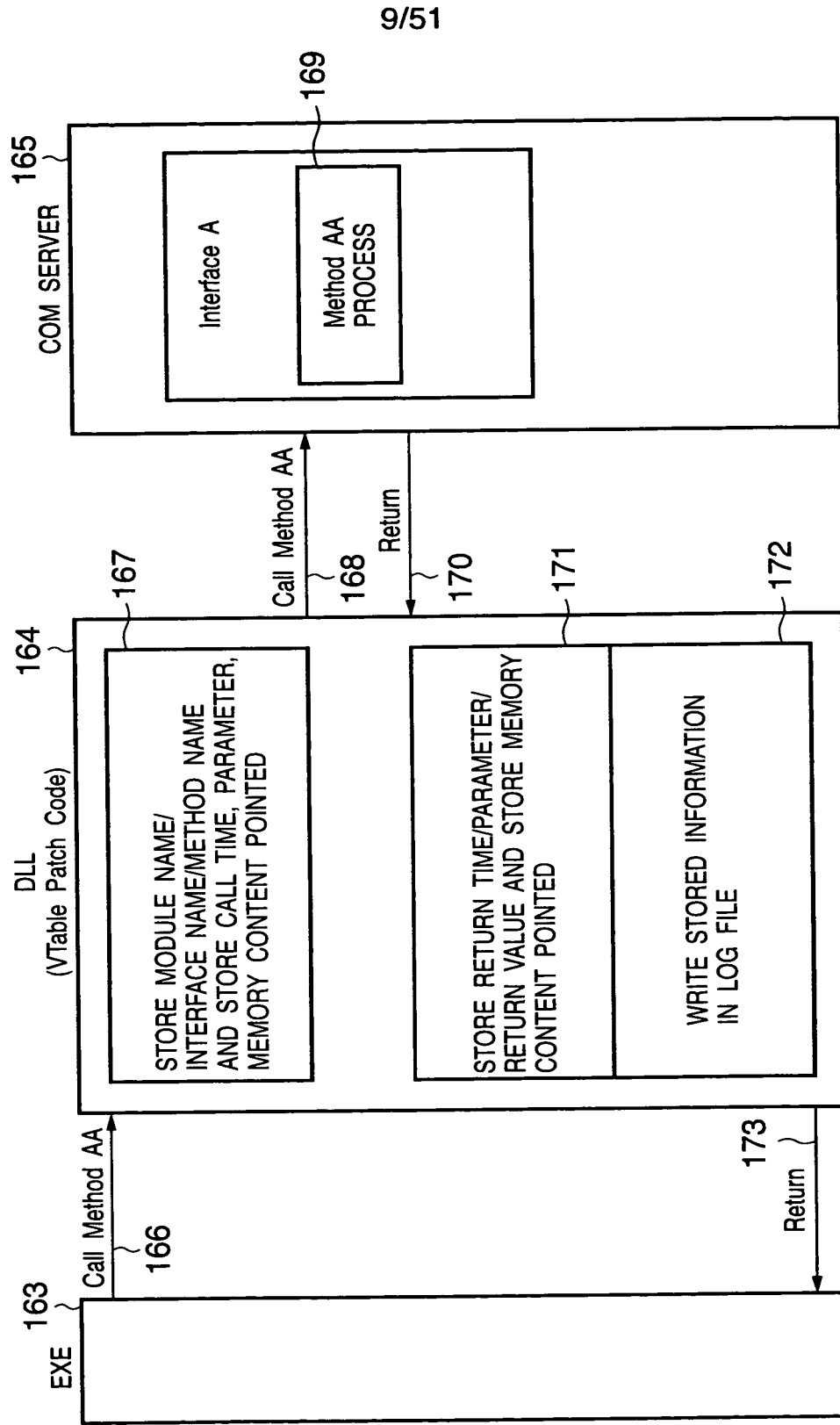


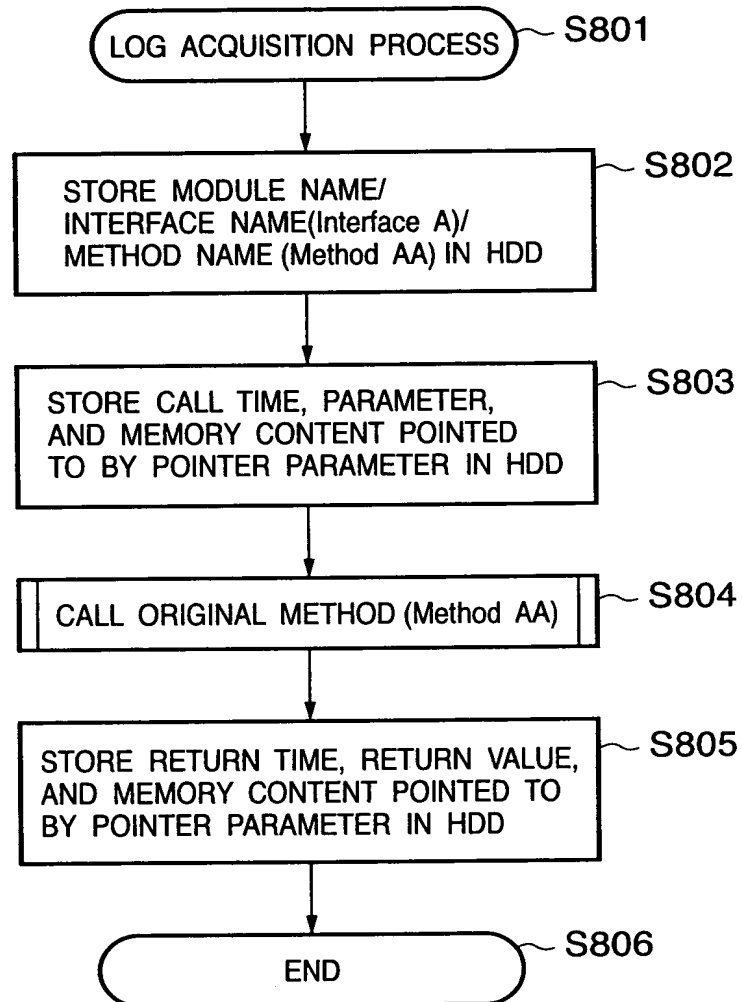
FIG. 8B

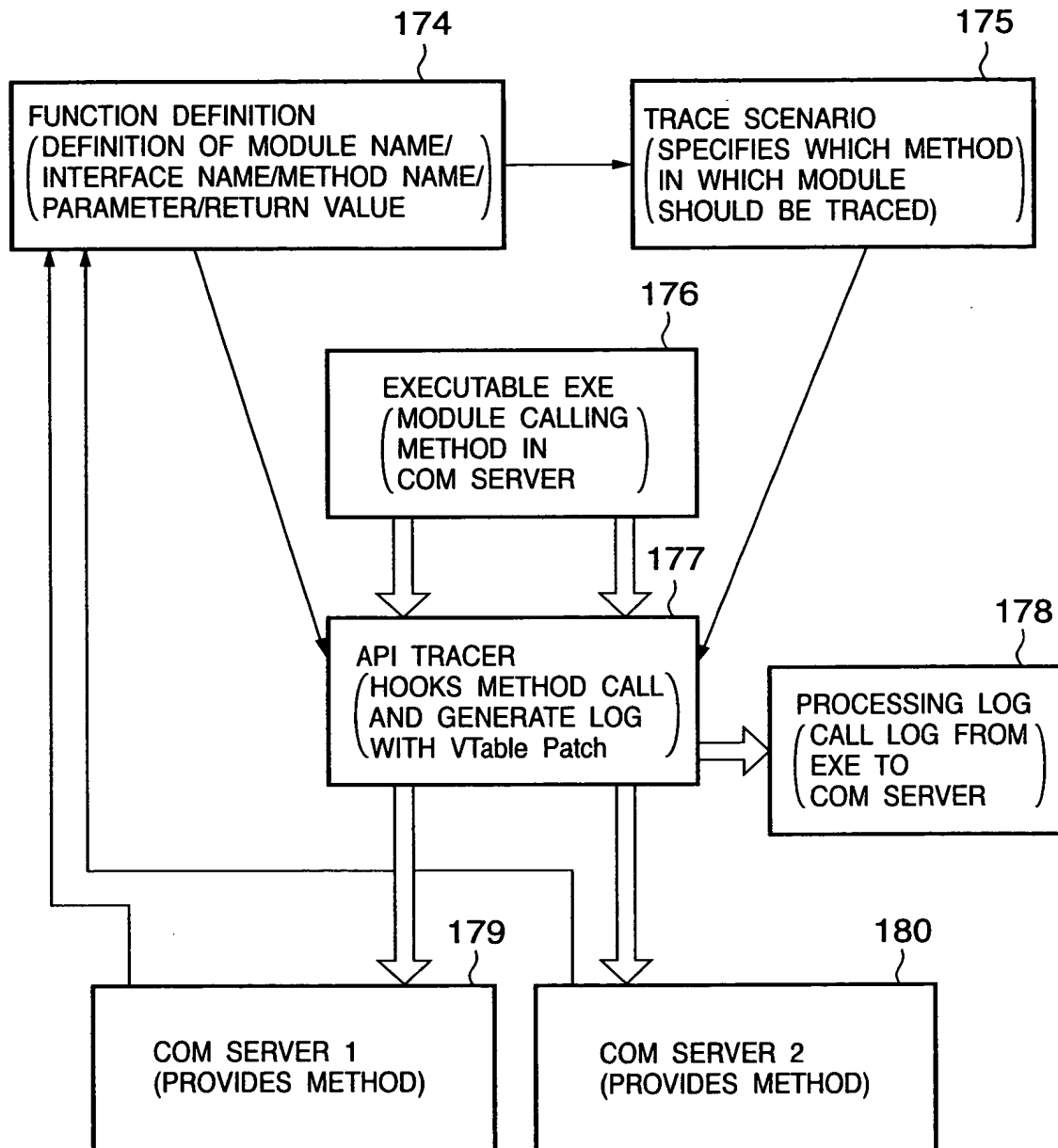
FIG. 9

FIG. 10

```

[
    uuid(58DB5633-0694-4340-97CE-4E1AC6BFFBA7),    //TestDIISd
    helpstring("TestDIISd Type Library For PAT"),
    version(1,0)
]

library TestDIISd

    typedef [public] struct
    {
        char chParam;
        unsigned char uchParam;
        short sParam;
        unsigned short usParam;
        int nParam;
        unsigned int unParam;
        long lParam;
        unsigned long ulParam;
        double dbParam;
        float fParam;
    }TESTSTRUCT;
    typedef [public] TESTSTRUCT *LPTESTSTRUCT;
//DEFINE_GUID(GUID_PROGID, 0x8e037d65, 0xefa0, 0x40e7, 0x91, 0x43, 0xef, 0x70, 0x56,
0x94, 0x5b, 0x79);
[
    uuid(8E037D65-EFA0-40e7-9143-EF7056945B79),
] helpstring("TestDIISd.dll for PAT object"),

    interface
    test
    {
        char_stdcall FuncCharStd([in] char chParam);
        char*_stdcall FuncPCharStd([in, out] char* lpchParam);

        TESTSTRUCT_stdcall FuncStructStd([in]TESTSTRUCT TestStruct);
        LPTESTSTRUCT_stdcall FuncPStructStd([in, out]LPTESTSTRUCTlp TestStruct);
    }
}

```

FIG. 11

200

```

#define PAT_PARAM_ATTR_ID 00000000-0000-0000-0000-000000000000

interface
test
{
    void_stdcall FuncBinids
    (
        [out, custom(PAT_PARAM_ATTR_ID, "binid_is()")] long* lpiParam
    );
    void_stdcall FuncSizels
    (
        [in] DWORD dwCount,
        [out, custom(PAT_PARAM_ATTR_ID, "sizeis_is(dwCount)")] int* lpnParam
    );
    void_stdcall FuncLengthls
    (
        [in] DWORD dwLength,
        [in, custom(PAT_PARAM_ATTR_ID, "lentgth_is(dwLength)")] char* lpszParam
    );
    void_stdcall FuncBytesls
    (
        [in] DWORD dwSize,
        [in, custom(PAT_PARAM_ATTR_ID, "bytes_is(dwSize)")] void* lpParam
    );
    void_stdcall FuncBytesls2
    (
        [out, custom(PAT_PARAM_ATTR_ID, "bytes_is(12)")] void* lpParam
    );
};

```

201

202

203

204

205

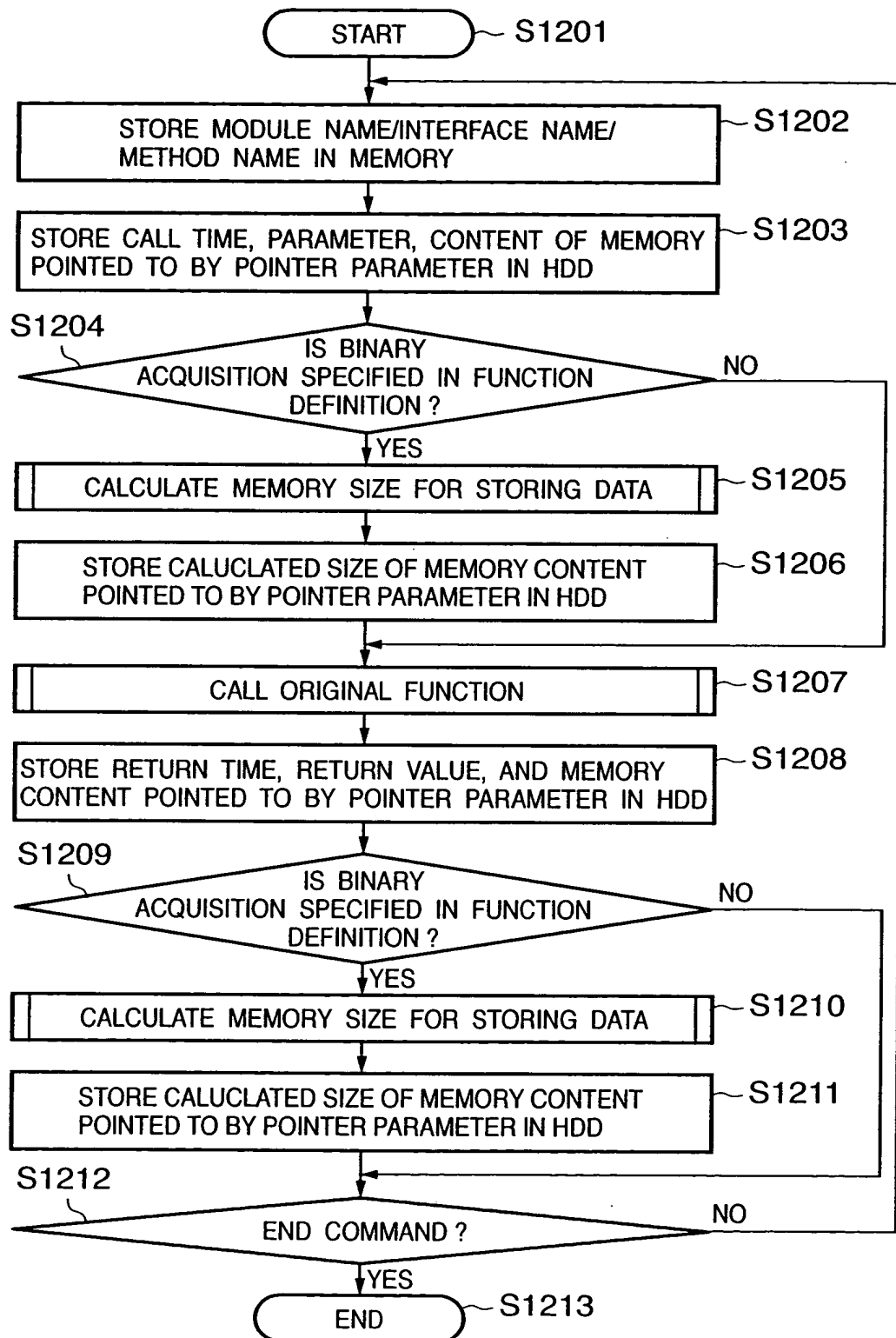
FIG. 12

FIG. 13

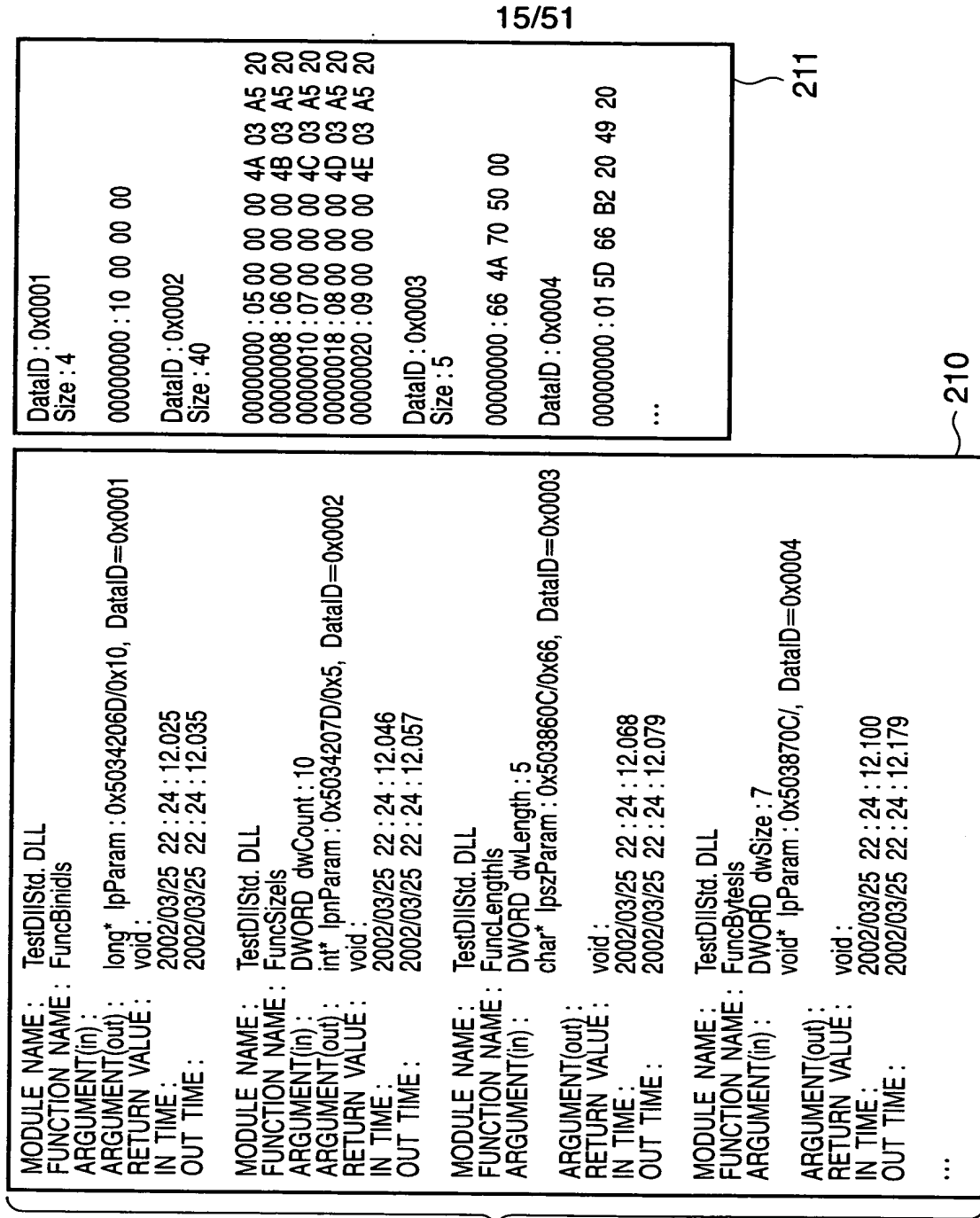


FIG. 14

```

#define PAT_PARAM_ATTR_ID 00000000-0000-0000-000000000000

typedef [public] struct ~ 220
{
    [in, custom(PAT_PARAM_ATTR_ID, "funcname_is(FuncInternal1)")] DWORD pfnFuncInternal1;
    [in, custom(PAT_PARAM_ATTR_ID, "funcname_is(FuncInternal2)")] DWORD pfnFuncInternal2;
    [in, custom(PAT_PARAM_ATTR_ID, "funcname_is(FuncInternal3)")] DWORD pfnFuncInternal3;
    [in, custom(PAT_PARAM_ATTR_ID, "funcname_is(FuncInternal4)")] DWORD pfnFuncInternal4;
}FUNCPOINTERARRAY;

interface
test
{
    void_stdcall SetCallBack ~ 221
    (
        [in, custom(PAT_PARAM_ATTR_ID, "funcname_is(FuncCallBack)")] DWORD
pfnFuncCallBack
    );
    void FuncCallBack([in] int nParam); ~ 222

    void_stdcall GetFuncPointer ~ 223
    (
        [out, custom(PAT_PARAM_ATTR_ID, "funcname_is(FuncInternal)")] DWORD
pfnFuncInternal
    );
    void FuncInternal([in, out] char* lpszParam); ~ 224

    void_stdcall GetFuncPointerArray
    (
        [out]FUNCPOINTERARRAY* pFuncPointerArray; ~ 225
    );
    void FuncInternal1([in] int nParam);
    void FuncInternal2([in, out] char* lpzaParam);
    void FuncInternal3([out] DWORD* dwParam); ~ 226
    void FuncInternal4();
};

```

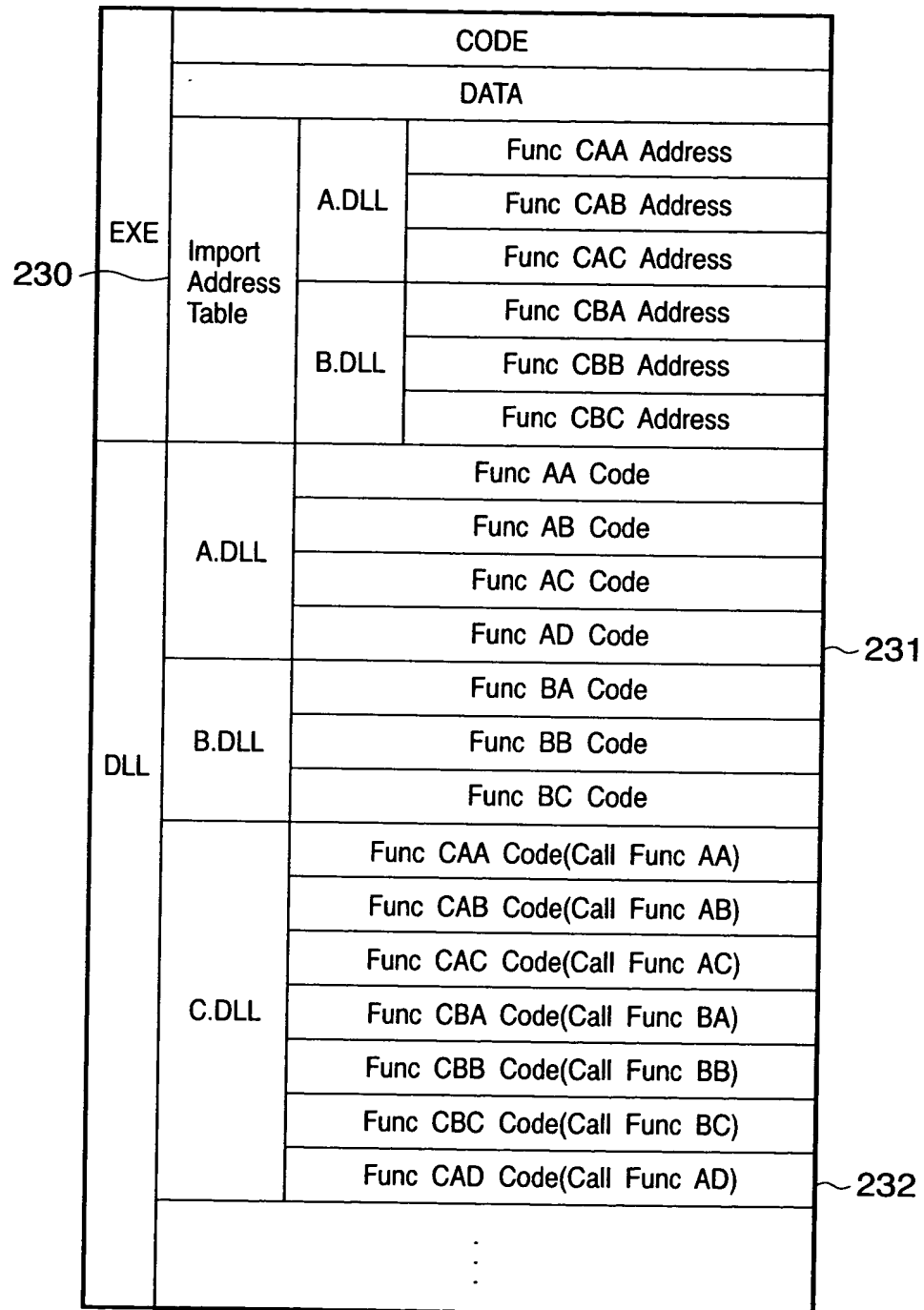

FIG. 15

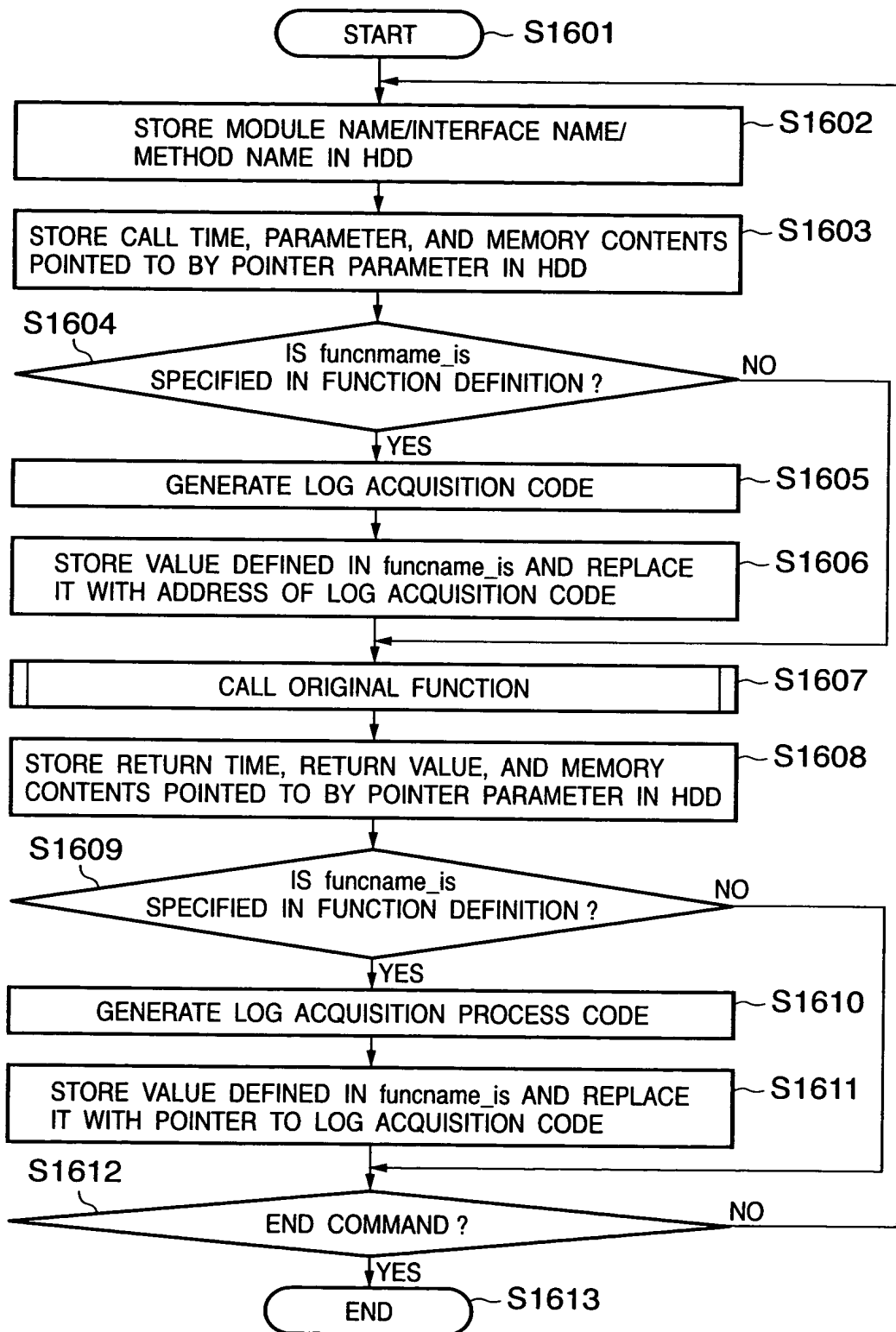
FIG. 16

FIG. 17

MODULE NAME :	TestDIIStd. DLL
FUNCTION NAME :	SetCallBack
ARGUMENT(in) :	DWORD pfnFuncCallBack : 0x0299103F
ARGUMENT(out) :	
RETURN VALUE :	void :
IN TIME :	2002/03/25 22 : 24 : 12.025
OUT TIME :	2002/03/25 22 : 24 : 12.035
MODULE NAME :	TestDIIStd. DLL
FUNCTION NAME :	GetFuncPointer
ARGUMENT(in) :	
ARGUMENT(out) :	DWORD pfnFuncInternal : 0x29913dF
RETURN VALUE :	void :
IN TIME :	2002/03/25 22 : 24 : 12.046
OUT TIME :	2002/03/25 22 : 24 : 12.057
MODULE NAME :	TestDIIStd. DLL
FUNCTION NAME :	GetFuncPointerArray
ARGUMENT(in) :	
ARGUMENT(out) :	FUNCPOINTERARRAY* pFuncPointerArray : 0x503860C DWORD FUNCPOINTERARRAY. pfnFuncInternal1 : 0x02997670 DWORD FUNCPOINTERARRAY. pfnFuncInternal2 : 0x02997708 DWORD FUNCPOINTERARRAY. pfnFuncInternal3 : 0x029977BE DWORD FUNCPOINTERARRAY. pfnFuncInternal4 : 0x0299784F
RETURN VALUE :	void :
IN TIME :	2002/03/25 22 : 24 : 12.068
OUT TIME :	2002/03/25 22 : 24 : 12.079

FIG. 18

```

MODULE NAME : TestDIIStd. DLL
FUNCTION NAME : SetCallBack
ARGUMENT(in) : DWORD pfnFuncCallBack : 0x0299103F
ARGUMENT(out) :
RETURN VALUE : void :
IN TIME : 2002/03/25 22 : 24 : 12.025
OUT TIME : 2002/03/25 22 : 24 : 12.035

MODULE NAME : TestDIIStd. DLL
FUNCTION NAME : FuncCallBack
ARGUMENT(in) : int nParam : 100
ARGUMENT(out) :
RETURN VALUE : void :
IN TIME : 2002/03/25 22 : 24 : 12.036
OUT TIME : 2002/03/25 22 : 24 : 12.040

MODULE NAME : TestDIIStd. DLL
FUNCTION NAME : GetFuncPointer
ARGUMENT(in) :
ARGUMENT(out) : DWORD pfnFuncInternal : 0x029913dF
RETURN VALUE : void :
IN TIME : 2002/03/25 22 : 24 : 12.046
OUT TIME : 2002/03/25 22 : 24 : 12.057

MODULE NAME : TestDIIStd. DLL
FUNCTION NAME : FuncInternal1
ARGUMENT(in) : char* lpszParam : 0x5038600/0
ARGUMENT(out) : char* lpszParam : 0x5038600/- 12
RETURN VALUE : void :
IN TIME : 2002/03/25 22 : 24 : 12.060
OUT TIME : 2002/03/25 22 : 24 : 12.065

MODULE NAME : TestDIIStd. DLL
FUNCTION NAME : GetFuncPointArray
ARGUMENT(in) :
ARGUMENT(out) : FUNCPOINTERARRAY* pFuncPointerArray : 0x503860C
                                DWORD FUNCPOINTERARRAY. pfnFuncInternal1 : 0x02997670
                                DWORD FUNCPOINTERARRAY. pfnFuncInternal2 : 0x02997708
                                DWORD FUNCPOINTERARRAY. pfnFuncInternal3 : 0x029977BE
                                DWORD FUNCPOINTERARRAY. pfnFuncInternal4 : 0x0299784F
RETURN VALUE : void :
IN TIME : 2002/03/25 22 : 24 : 12.068
OUT TIME : 2002/03/25 22 : 24 : 12.079

MODULE NAME : TestDIIStd. DLL
FUNCTION NAME : FuncInternal4
ARGUMENT(in) :
ARGUMENT(out) :
RETURN VALUE : void :
IN TIME : 2002/03/25 22 : 24 : 12.080
OUT TIME : 2002/03/25 22 : 24 : 12.099

```

FIG. 19

```
#define PAT_PARAM_ATTR_ID 00000000-0000-0000-000000000000

interface
test
{
    void_stdcall FuncArrayIs
    (
        [in] DWORD dwCount,
        [in, out, custom(PAT_PARAM_ATTR_ID, "array_is(dwCount)")] int* lpnParam
    );
};
```

240

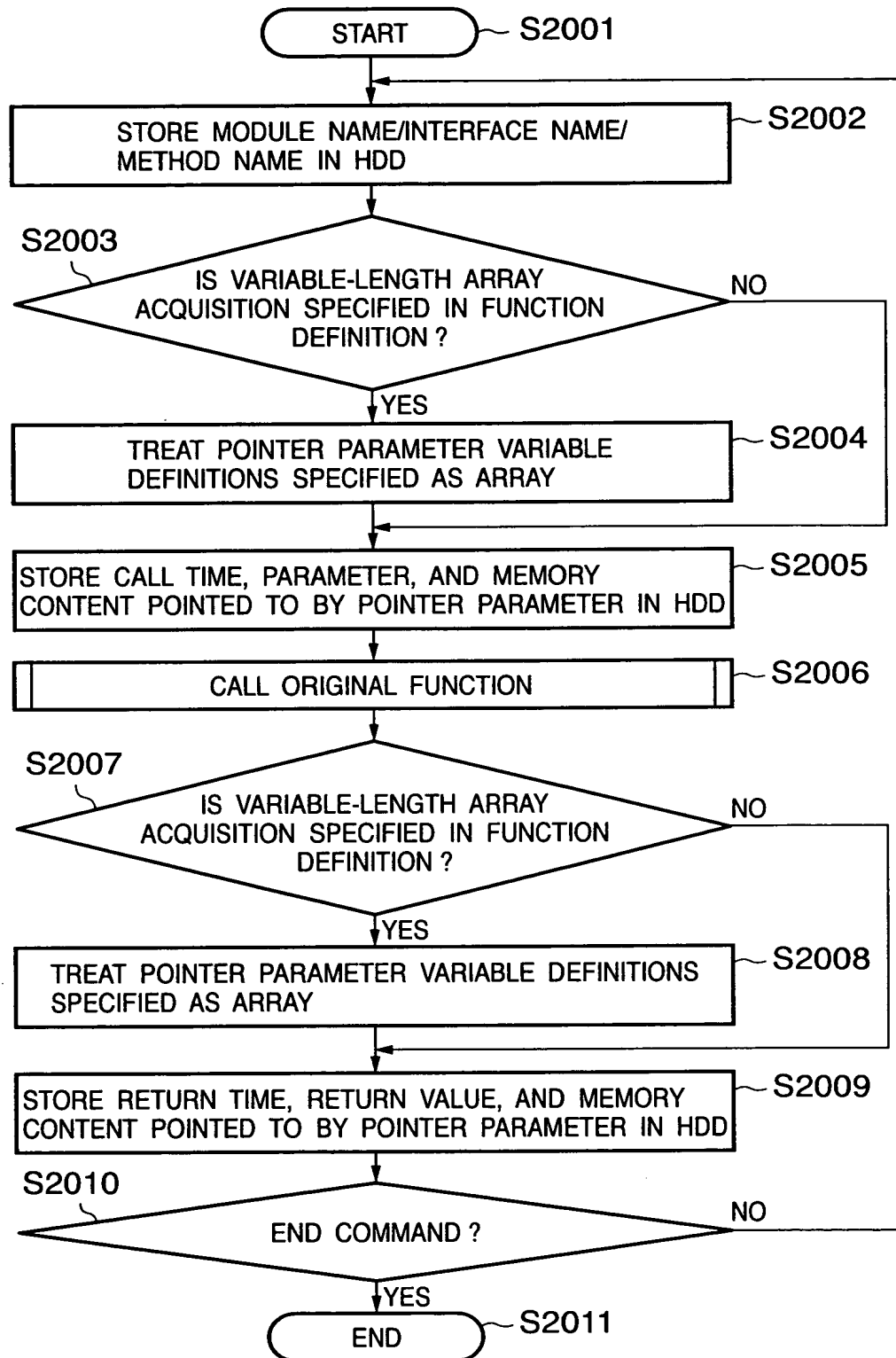
FIG. 20

FIG. 21

MODULE NAME : TestDIIStd. DLL
 FUNCTION NAME : FuncArrayls
 ARGUMENT(in) : DWORD dwCount : 4
 int* lpnParam : 0x5034206D/0x00
 ARGUMENT(out) : int* lpnParam : 0x5034206D/0x01
 RETURN VALUE : void :
 IN TIME : 2002/03/25 22 : 24 : 12.025
 OUT TIME : 2002/03/25 22 : 24 : 12.035

250

MODULE NAME : TestDIIStd. DLL
 FUNCTION NAME : FuncArrayls
 ARGUMENT(in) : DWORD dwCount : 3
 int* lpnParam : 0x5034207D/0x00
 ARGUMENT(out) : int* lpnParam : 0x5034207D/0x05
 RETURN VALUE : void :
 IN TIME : 2002/03/25 22 : 24 : 12.046
 OUT TIME : 2002/03/25 22 : 24 : 12.057

...

251

MODULE NAME : TestDIIStd. DLL
 FUNCTION NAME : FuncArrayls
 ARGUMENT(in) : DWORD dwCount : 4
 int* lpnParam : 0x5034206D/Array (int : 0 : 0x00, int : 1 : 0x00, int : 2 : 0x00, int : 3 : 0x00)
 ARGUMENT(out) : int* lpnParam : 0x5034206D/Array (int : 0 : 0x01, int : 1 : 0x02, int : 2 : 0x03, int : 3 : 0x04)
 RETURN VALUE : void :
 IN TIME : 2002/03/25 22 : 24 : 12.025
 OUT TIME : 2002/03/25 22 : 24 : 12.035

MODULE NAME : TestDIIStd. DLL
 FUNCTION NAME : FuncArrayls
 ARGUMENT(in) : DWORD dwCount : 3
 int* lpnParam : 0x5034207D/Array (int : 0 : 0x00, int : 1 : 0x00, int : 2 : 0x00)
 ARGUMENT(out) : int* lpnParam : 0x5034207D/Array (int : 0 : 0x05, int : 1 : 0x10, int : 2 : 0x15)
 RETURN VALUE : void :
 IN TIME : 2002/03/25 22 : 24 : 12.046
 OUT TIME : 2002/03/25 22 : 24 : 12.057

...

FIG. 22

```
typedef struct
{
    DWORD dwSize ;
    DWORD dwParam1 ;
    DWORD dwParam2 ;
    DWORD dwParam3 ;
}STRUCTSIZE1 ;

typedef struct
{
    DWORD dwSize ;
    DWORD dwParam1 ;
    DWORD dwParam2 ;
    DWORD dwParam3 ;
    DWORD dwParam4 ;
}STRUCTSIZE2 ;

typedef struct
{
    DWORD dwSize ;
    DWORD dwParam1 ;
    DWORD dwParam2 ;
    DWORD dwParam3 ;
    DWORD dwParam4 ;
    DWORD dwParam5 ;
}STRUCTSIZE3 ;

void FuncGetData (DWORD dwKind, void* lpBuf)
{
    switch(dwKind)
    {
        case 1 :
            //lpBuf IS TREATED AS THE POINTER TO STRUCTSIZE1
            break;
        case 2 :
            //lpBuf IS TREATED AS THE POINTER TO STRUCTSIZE2
            break;
        case 3 :
            //lpBuf IS TREATED AS THE POINTER TO STRUCTSIZE3
            break;
    }
}
```

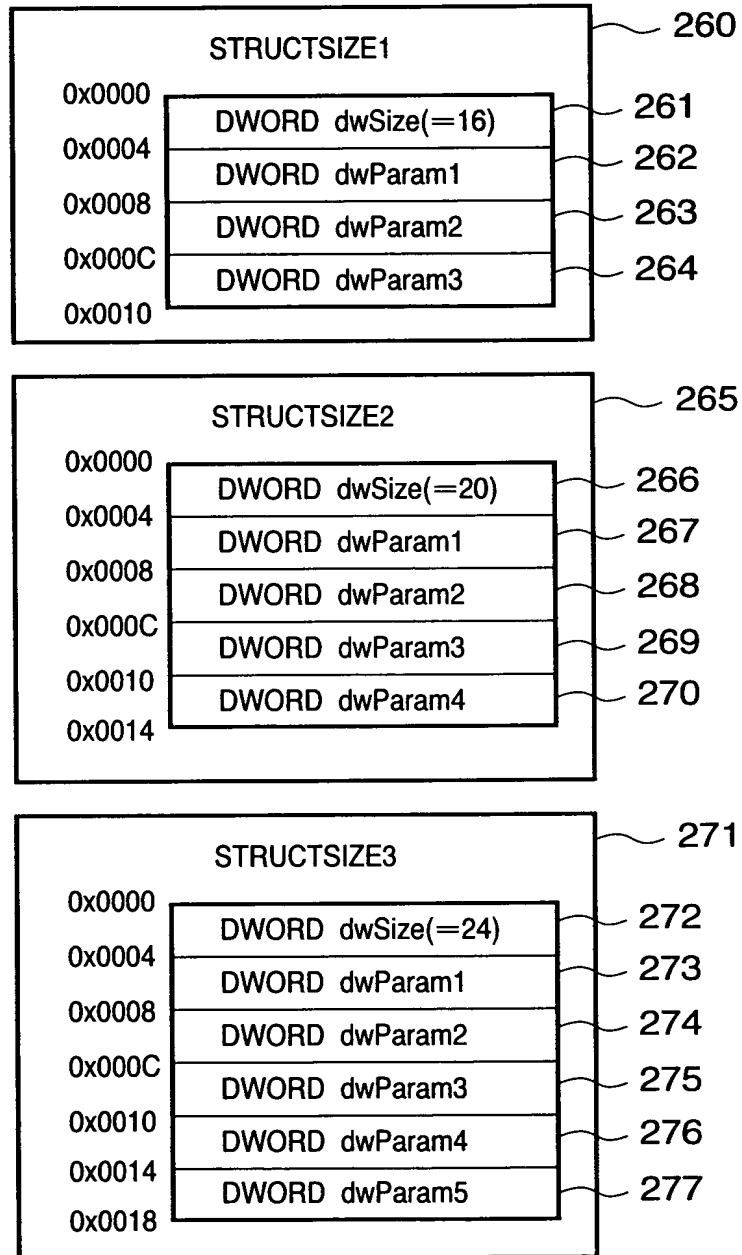

FIG. 23

FIG. 24

```

#define PAT_PARAM_ATTR_ID 00000000-0000-0000-0000-000000000000

typedef [public] struct
{
    [custom (PAT_PARAM_ATTR_ID, "structsize_is()")]DWORD dwSize;
    DWORD dwParam1 ;
    DWORD dwParam2 ;
    DWORD dwParam3 ;
    DWORD dwParam4 ;
    DWORD dwParam5 ;

}STRUCTSIZE ; ~ 291

interface
test
{
    void FuncGetData
    (
        [in] DWORD dwKind,
        [in, out] STRUCTSIZE* lpBuf
    );

};

```

290

292

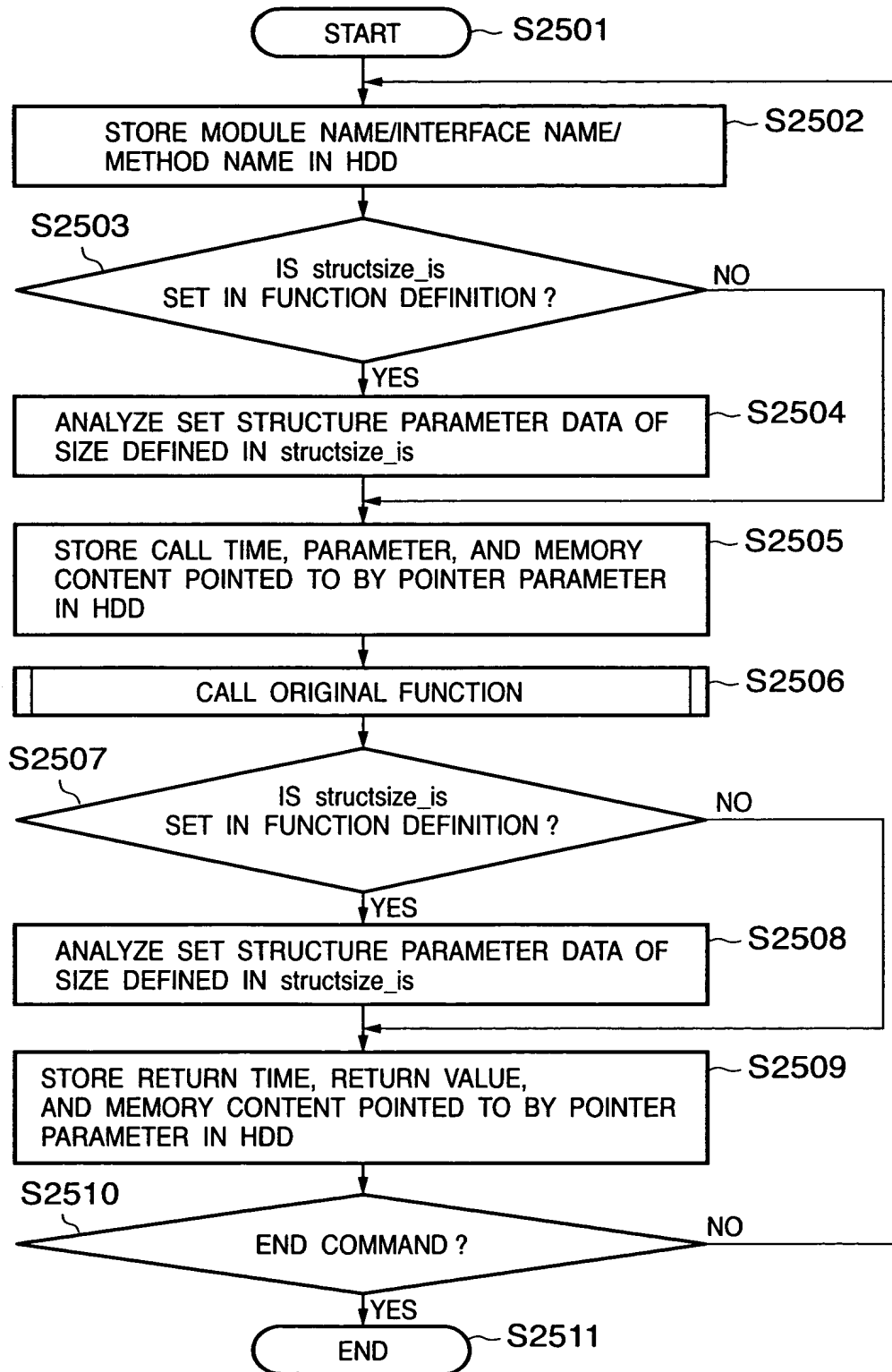
FIG. 25

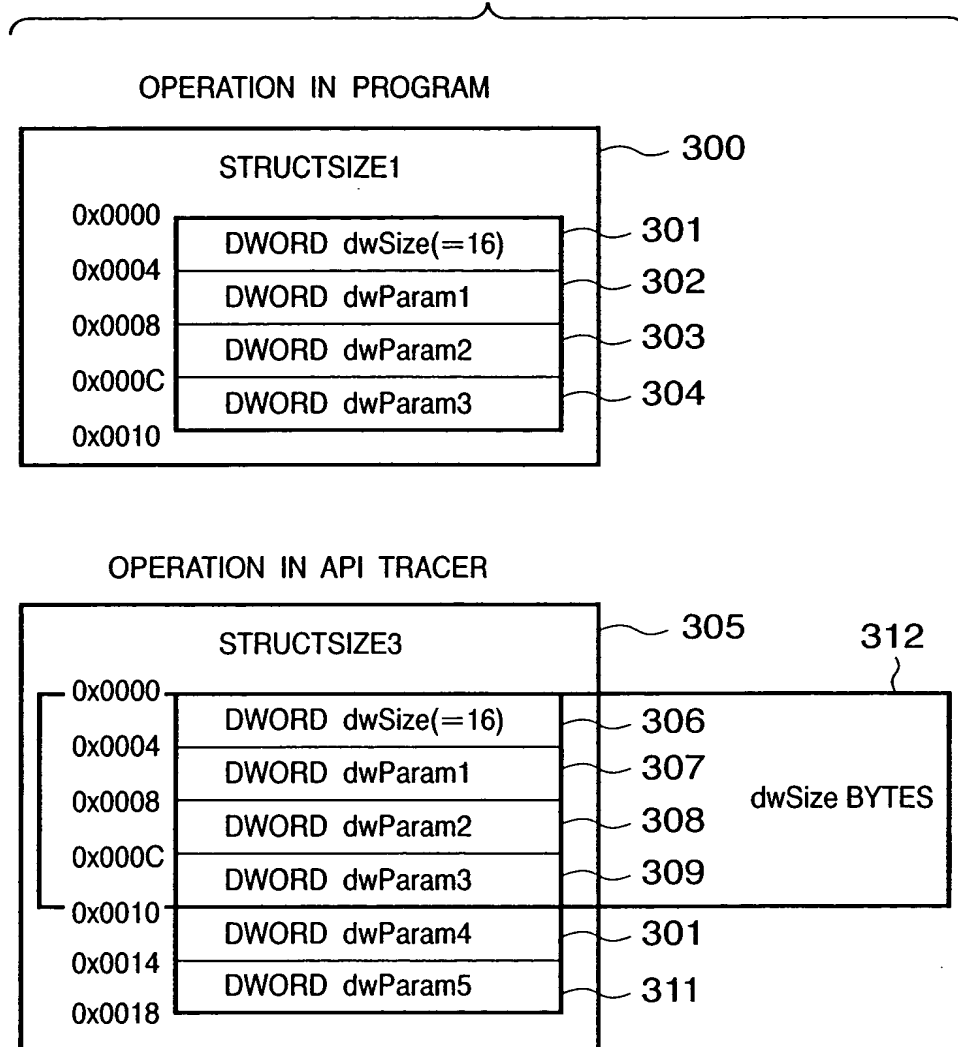
FIG. 26

FIG. 27

```

MODULE NAME : TestDIIStd. DLL
FUNCTION NAME : FuncGetData
ARGUMENT(in) : DWORD dwKind : 1
                STRUCTSIZE* pBuf : 0x503860C
                DWORD STRUCTSIZE. dwSize : 16
                DWORD STRUCTSIZE. dwParam1 : 0
                DWORD STRUCTSIZE. dwParam2 : 0
                DWORD STRUCTSIZE. dwParam3 : 0
ARGUMENT(out) : STRUCTSIZE* pBuf : 0x503860C
                DWORD STRUCTSIZE. dwSize : 16
                DWORD STRUCTSIZE. dwParam1 : 1
                DWORD STRUCTSIZE. dwParam2 : 2
                DWORD STRUCTSIZE. dwParam3 : 3
RETURN VALUE : void :
IN TIME : 2002/03/25 22 : 24 : 12.025
OUT TIME : 2002/03/25 22 : 24 : 12.035

MODULE NAME : TestDIIStd. DLL
FUNCTION NAME : FuncGetData
ARGUMENT(in) : DWORD dwKind : 3
                STRUCTSIZE* pBuf : 0x503990C
                DWORD STRUCTSIZE. dwSize : 24
                DWORD STRUCTSIZE. dwParam1 : 0
                DWORD STRUCTSIZE. dwParam2 : 0
                DWORD STRUCTSIZE. dwParam3 : 0
                DWORD STRUCTSIZE. dwParam4 : 0
                DWORD STRUCTSIZE. dwParam5 : 0
ARGUMENT(out) : STRUCTSIZE* pBuf : 0x503990C
                DWORD STRUCTSIZE. dwSize : 24
                DWORD STRUCTSIZE. dwParam1 : 10
                DWORD STRUCTSIZE. dwParam2 : 20
                DWORD STRUCTSIZE. dwParam3 : 30
                DWORD STRUCTSIZE. dwParam4 : 40
                DWORD STRUCTSIZE. dwParam5 : 50
RETURN VALUE : void :
IN TIME : 2002/03/25 22 : 24 : 12.046
OUT TIME : 2002/03/25 22 : 24 : 12.057

```

...

FIG. 28

```
typedef struct
{
    char chParam;
    DWORD dwParam ;
    short shParam ;
}STRUCTKIND1 ;

typedef struct
{
    short shParam ;
    DWORD dwParam ;
    char chParam;
}STRUCTKIND2 ;

typedef struct
{
    char chParam;
    DWORD dwParam ;
    short shParam ;
    long lParam ;
    int nParam ;
}STRUCTKIND3 ;

void FuncGetData (DWORD dwKind, void* lpBuf)
{
    switch(dwKind)
    {
    case 1 :
        //lpBuf IS TREATED AS THE POINTER TO STRUCTKIND1
        break;
    case 2 :
        //lpBuf IS TREATED AS THE POINTER TO STRUCTKIND2
        break;
    case 3 :
        //lpBuf IS TREATED AS THE POINTER TO STRUCTKIND3
        break;
    }
}
```

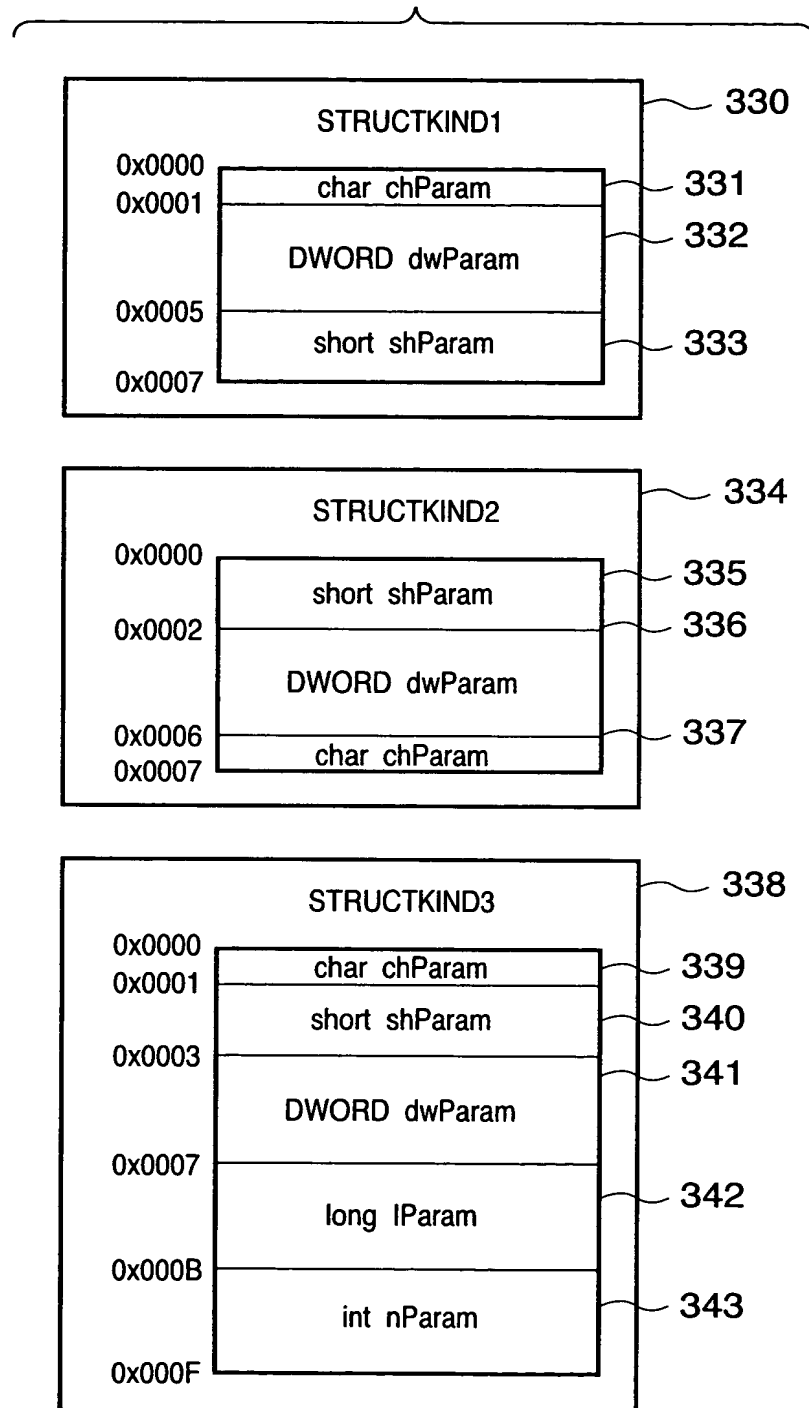
FIG. 29

FIG. 30

```

#define PAT_PARAM_ATTR_ID 00000000-0000-0000-000000000000

typedef [public] struct
{
    char chParam ;
    DWORD dwParam ;
    short shParam ;
}STRUCTKIND1 ;

typedef [public] struct
{
    short shParam ;
    DWORD dwParam ;
    char chParam ;
}STRUCTKIND2 ;

typedef [public] struct
{
    char chParam ;
    short shParam ;
    DWORD dwParam ;
    long lParam ;
    int nParam ;
}STRUCTKIND3 ;

interface
test
{
    void FuncGetData
    (
        [in] DWORD dwKind,
        [in, out, custom(PAT_PARAM_ATTR_ID,
        "structKind_is(dwKind : 1 : STRUCTKIND1*, 2 : STRUCTKIND2*, 3 : STRUCTKIND3* )")]
        void* lpBuf
    );
};

```

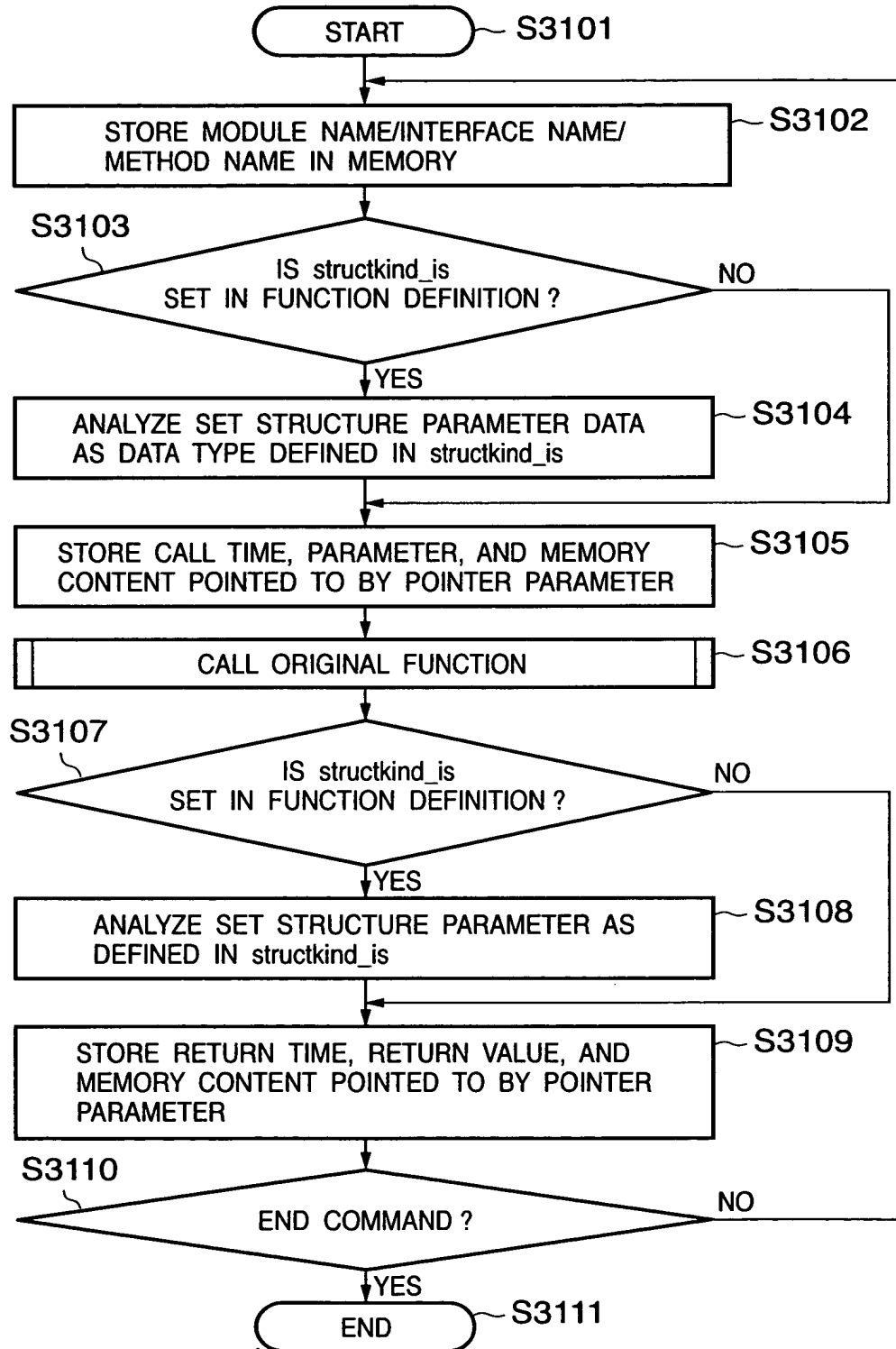

FIG. 31

FIG. 32

```

MODULE NAME :    TestDIISD. DLL
FUNCTION NAME :   FuncGetData
ARGUMENT(in) :    DWORD dwKind : 1
                  STRUCTKIND1* pBuf : 0x503860C
                  char STRUCTKIND1.chParam : 0
                  DWORD STRUCTKIND1.dwParam : 0
                  short STRUCTKIND1.shParam : 0
ARGUMENT(out) :   STRUCTKIND1* pBuf : 0x503860C
                  char STRUCTKIND1.chParam : 1
                  DWORD STRUCTKIND1.dwParam : 2
                  short STRUCTKIND1.shParam : 3

RETURN VALUE :    void :
IN TIME :          2002/03/25 22 : 24 : 12.025
OUT TIME :         2002/03/25 22 : 24 : 12.035

MODULE NAME :    TestDIISD. DLL
FUNCTION NAME :   FuncGetData
ARGUMENT(in) :    DWORD dwKind : 3
                  STRUCTKIND3* pBuf : 0x503990C
                  char STRUCTKIND3.chParam : 0
                  short STRUCTKIND3.shParam : 0
                  DWORD STRUCTKIND3.dwParam : 0
                  long STRUCTKIND3.lParam : 0
                  int STRUCTKIND3.nParam : 0
ARGUMENT(out) :   STRUCTKIND3* pBuf : 0x503990C
                  char STRUCTKIND3.chParam : 10
                  short STRUCTKIND3.shParam : 20
                  DWORD STRUCTKIND3.dwParam : 30
                  long STRUCTKIND3.lParam : 40
                  int STRUCTKIND3.nParam : 50

RETURN VALUE :    void :
IN TIME :          2002/03/25 22 : 24 : 12.046
OUT TIME :         2002/03/25 22 : 24 : 12.057

```

...

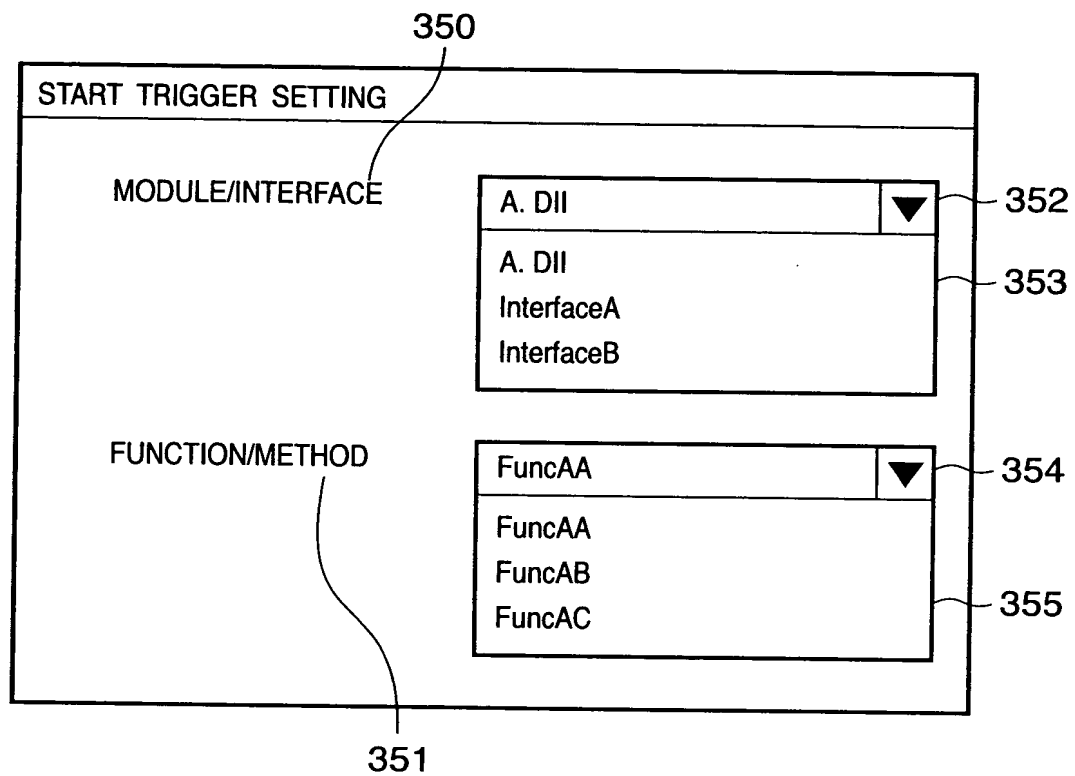
FIG. 33

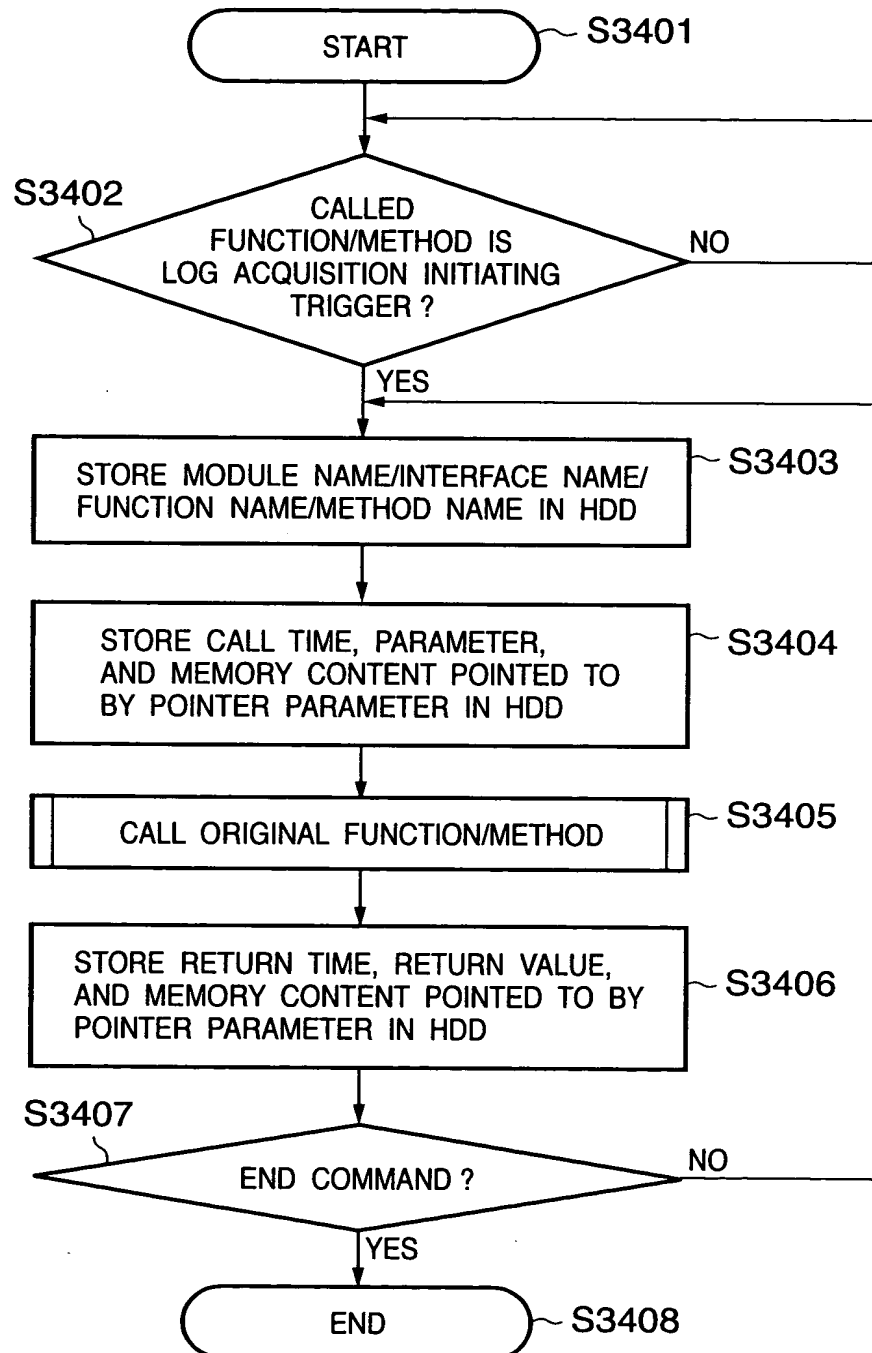
FIG. 34

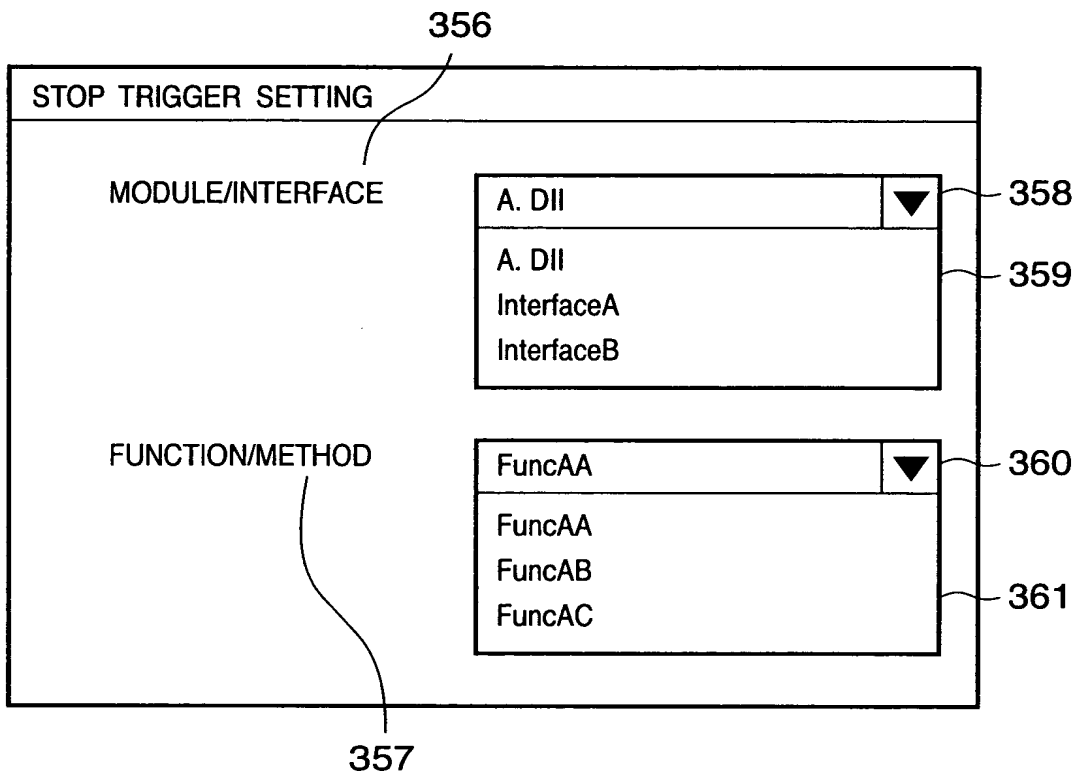
FIG. 35

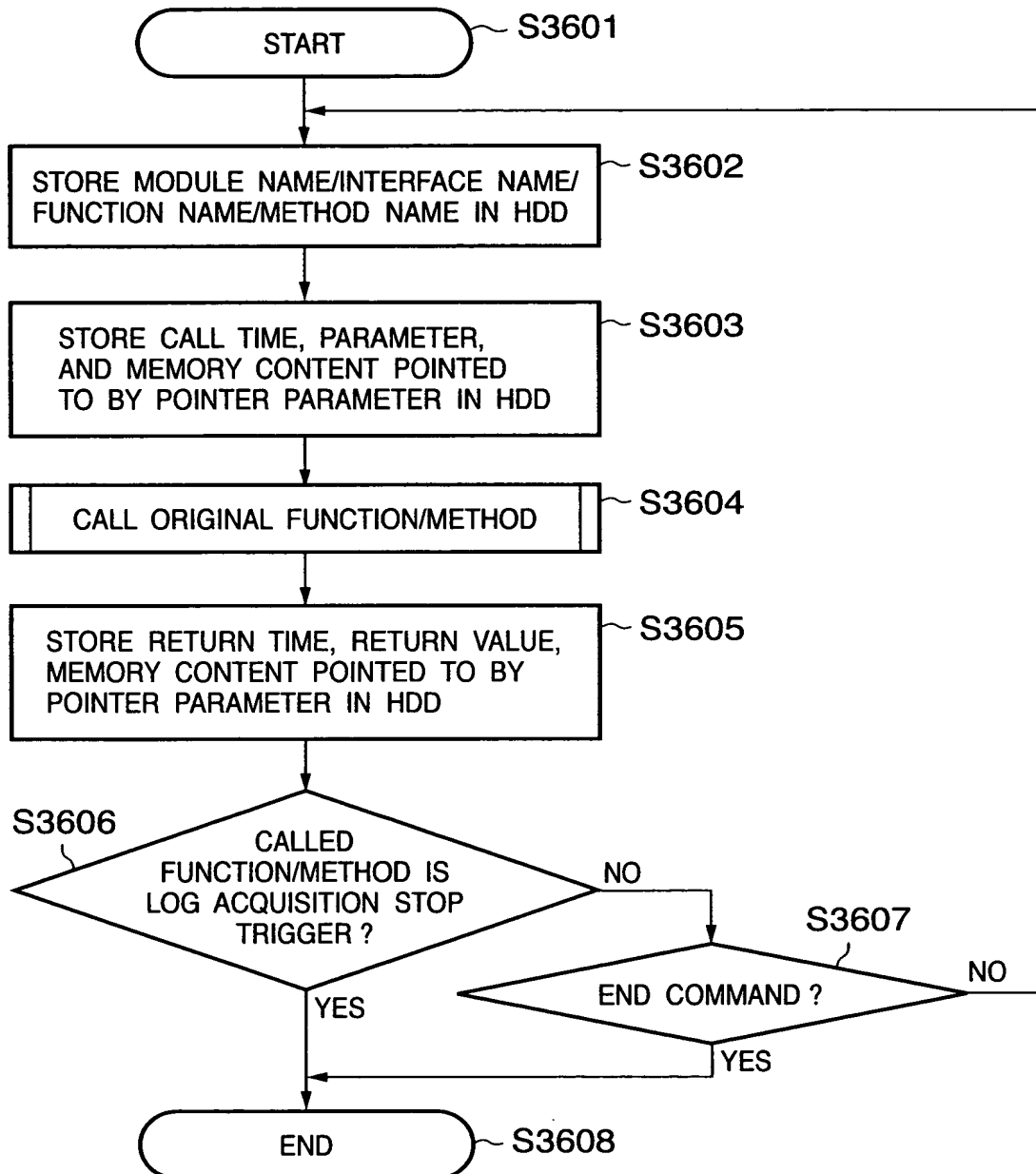
FIG. 36

FIG. 37

362

ERROR TRIGGER SETTING

MODULE/INTERFACE

364

A. Dll

365

A. Dll
InterfaceA
InterfaceB

FUNCTION/METHOD

363

366

FuncAA

367

FuncAA
FuncAB
FuncAC

☒ USE TRIGGER ONLY IF FUNCTION/METHOD IS ABORTED BY ERROR

368

FIG. 38

MODULE NAME :	A. DLL
FUNCTION NAME :	FuncAA
ARGUMENT :	DWORD dwID : Err>100
RETURN VALUE :	DWORD dwRet : Err==0
MODULE NAME :	A. DLL
FUNCTION NAME :	FuncAB
ARGUMENT :	DWORD dwHandle : Err==0
RETURN VALUE :	int nRet : Err<=-1
MODULE NAME :	B. DLL
INTERFACE NAME :	InterfaceA
METHOD NAME :	MethodAA
ARGUMENT :	DWORD dwID : Err>100
RETURN VALUE :	DWORD dwHandle: Err==0
MODULE NAME :	B. DLL
INTERFACE NAME :	InterfaceA
METHOD NAME :	MethodAB
ARGUMENT :	DWORD dwID : Err<=0
RETURN VALUE :	DWORD dwRet: Err!=0
MODULE NAME :	B. DLL
INTERFACE NAME :	InterfaceB
METHOD NAME :	MethodBA
ARGUMENT :	DWORD dwID : Err>=0
RETURN VALUE :	DWORD dwRet: Err!=0

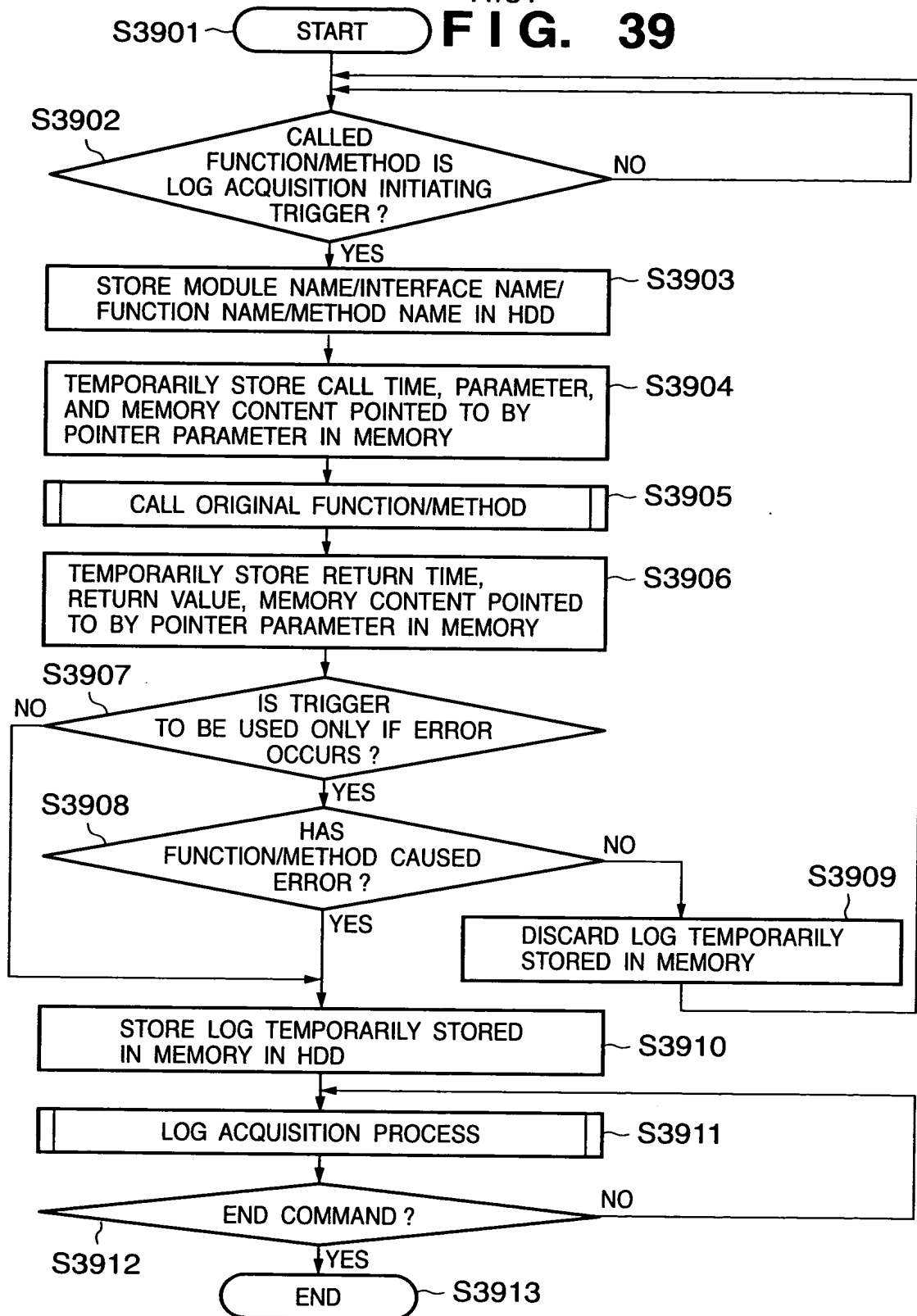


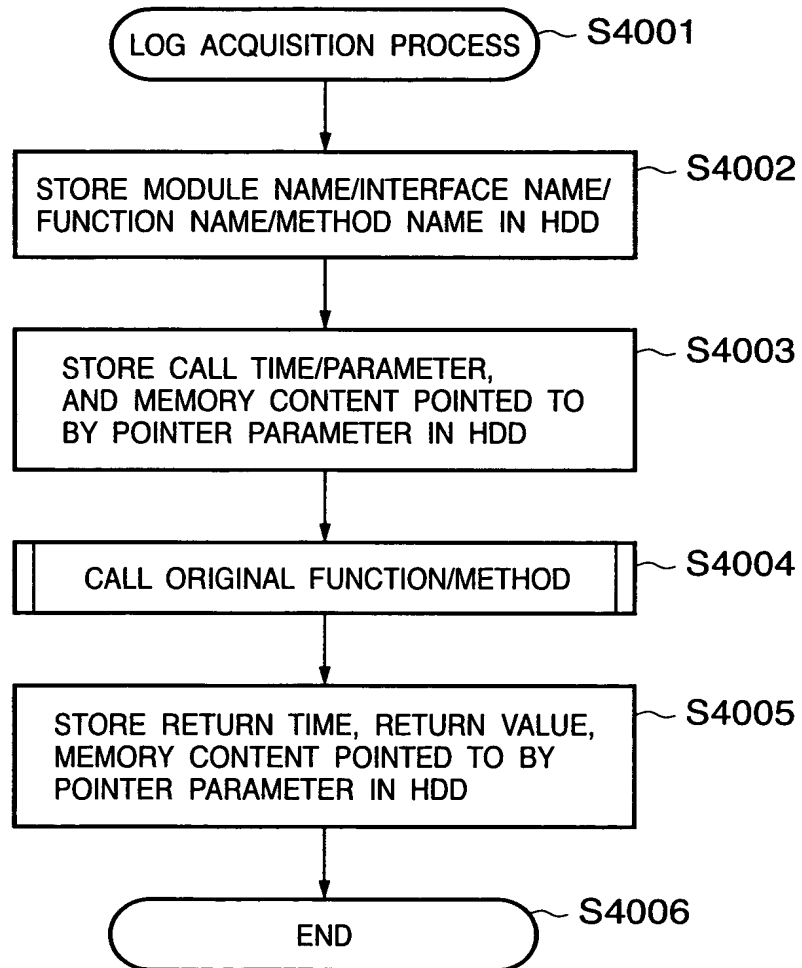
FIG. 40

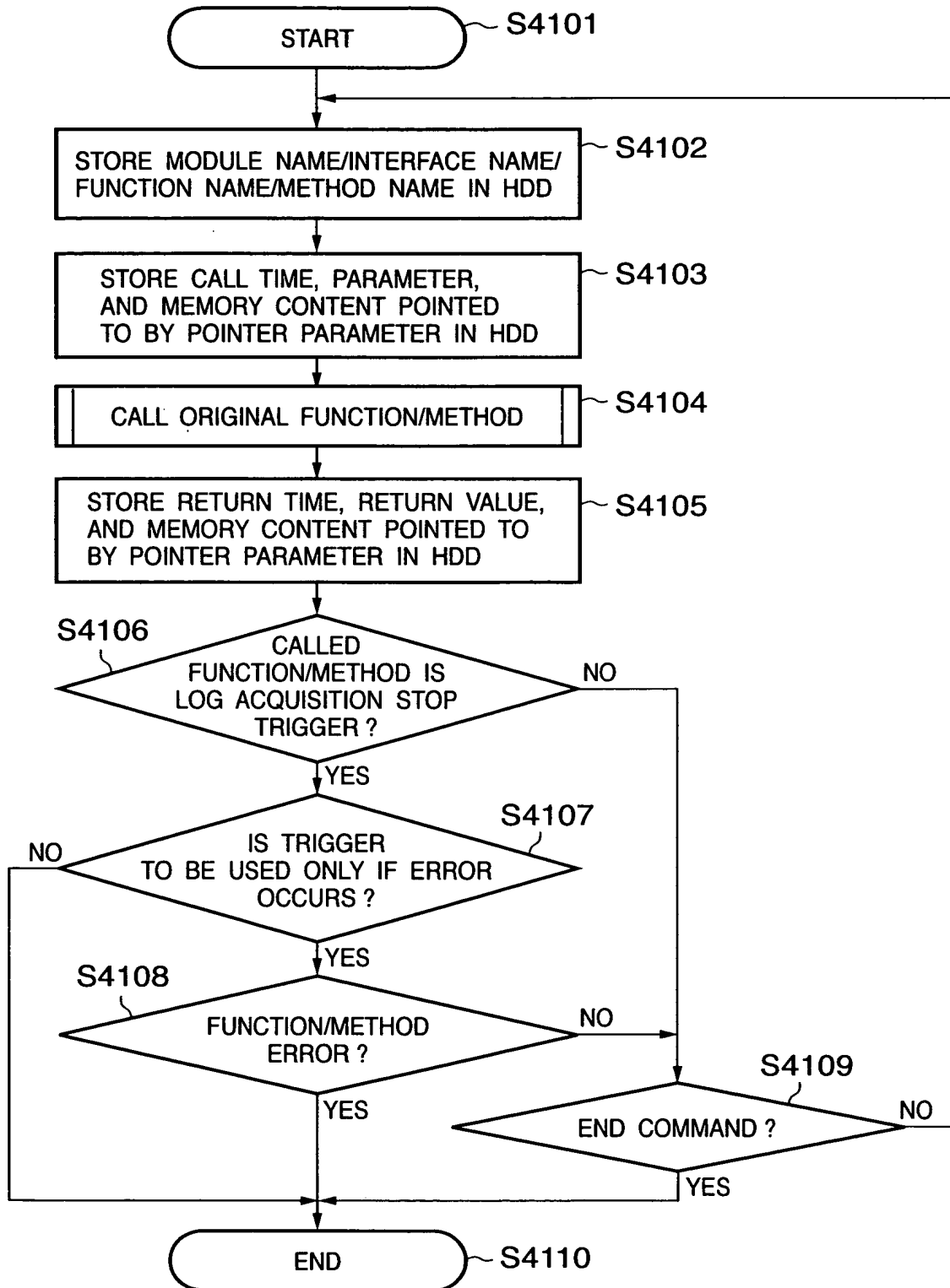
FIG. 41

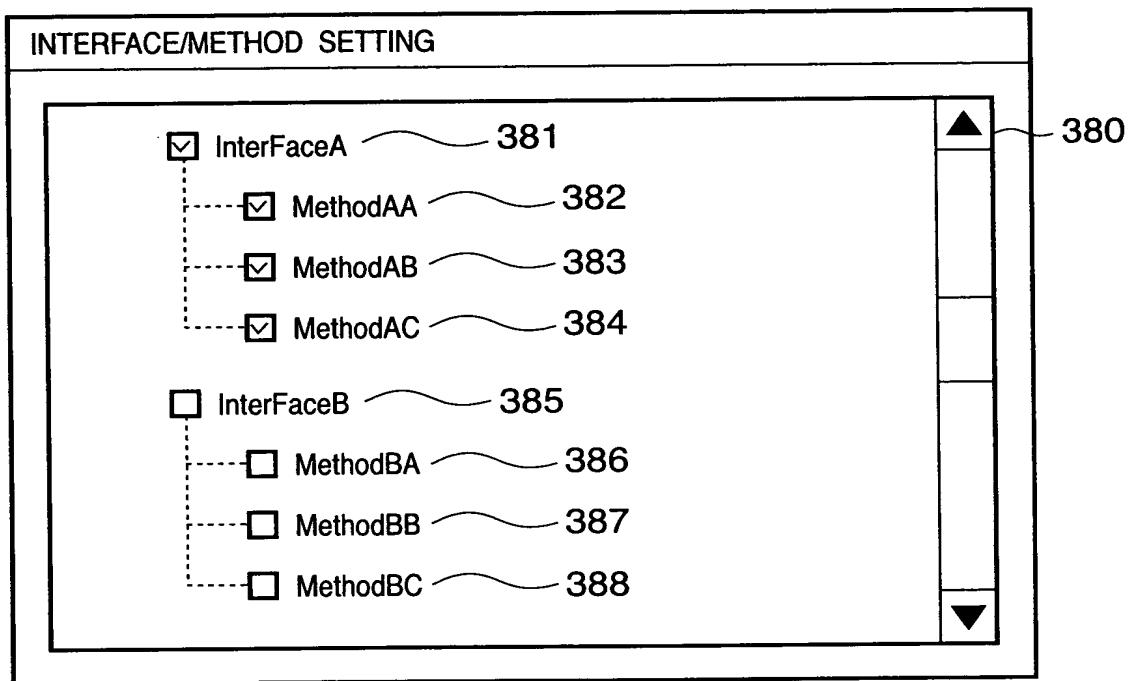
FIG. 42

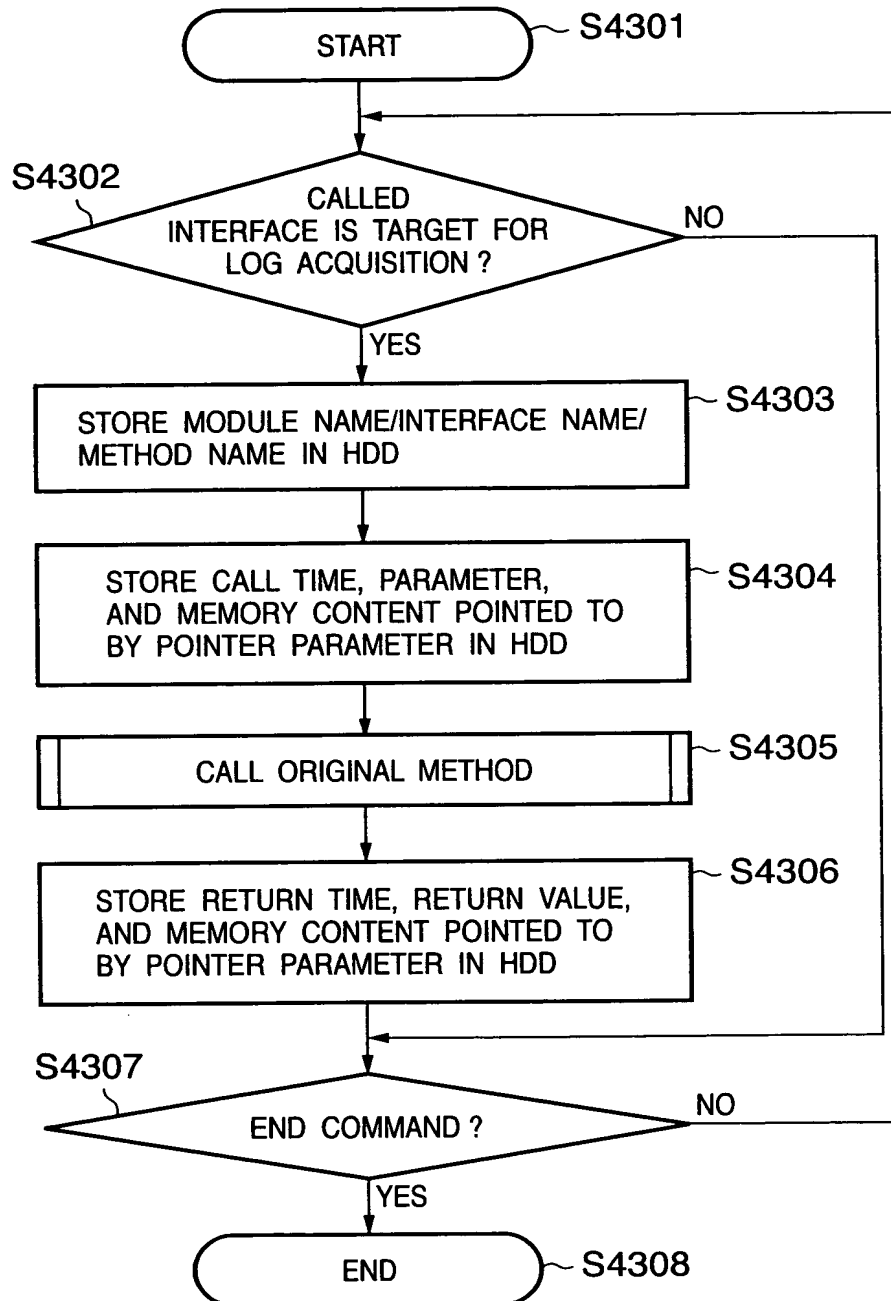
FIG. 43

FIG. 44

INTERFACE/METHOD SETTING

<input checked="" type="checkbox"/>	InterFaceA	390
<input checked="" type="checkbox"/>	MethodAA	391
<input type="checkbox"/>	MethodAB	392
<input checked="" type="checkbox"/>	MethodAC	393
<input type="checkbox"/>	InterFaceB	394
<input checked="" type="checkbox"/>	MethodBA	395
<input type="checkbox"/>	MethodBB	396
<input type="checkbox"/>	MethodBC	397

389

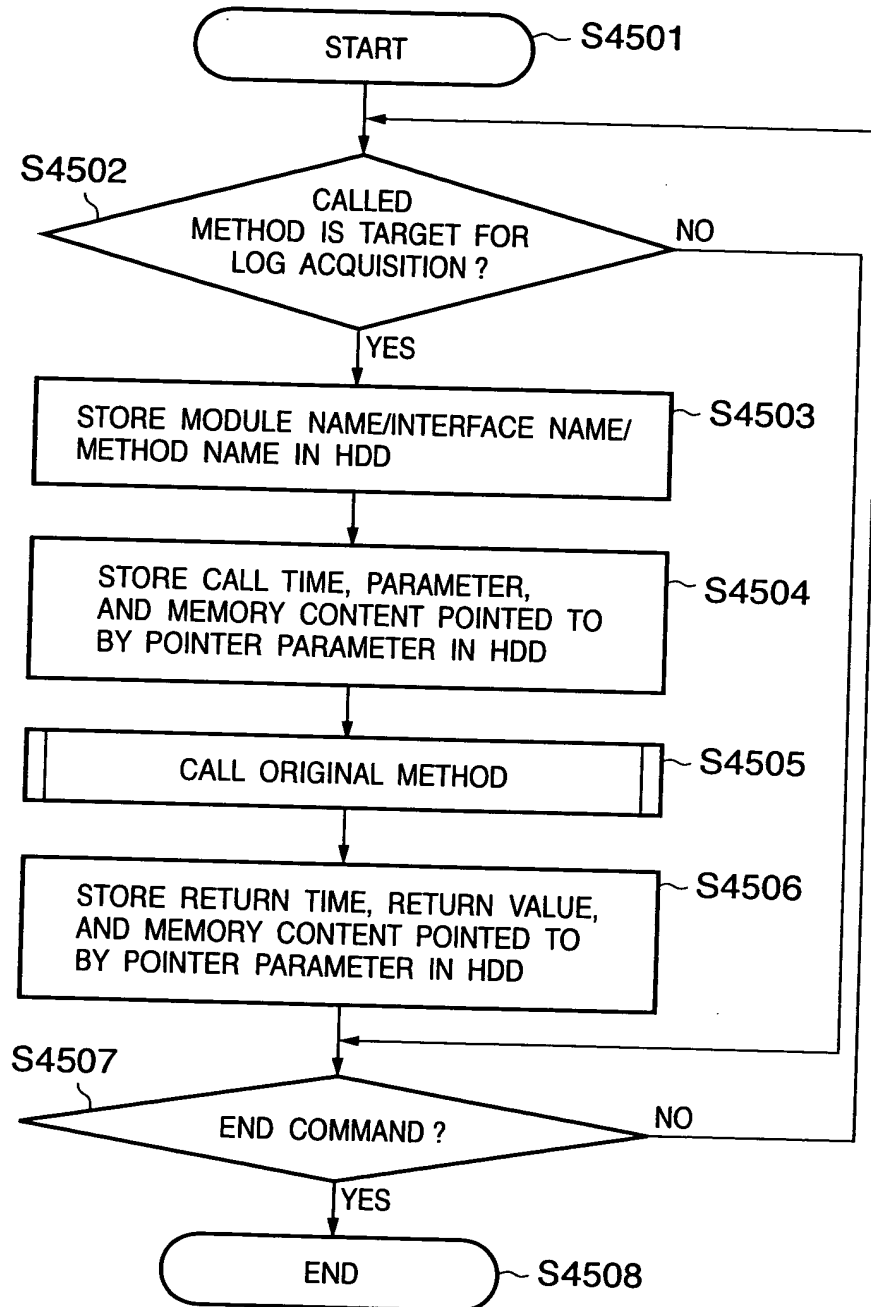
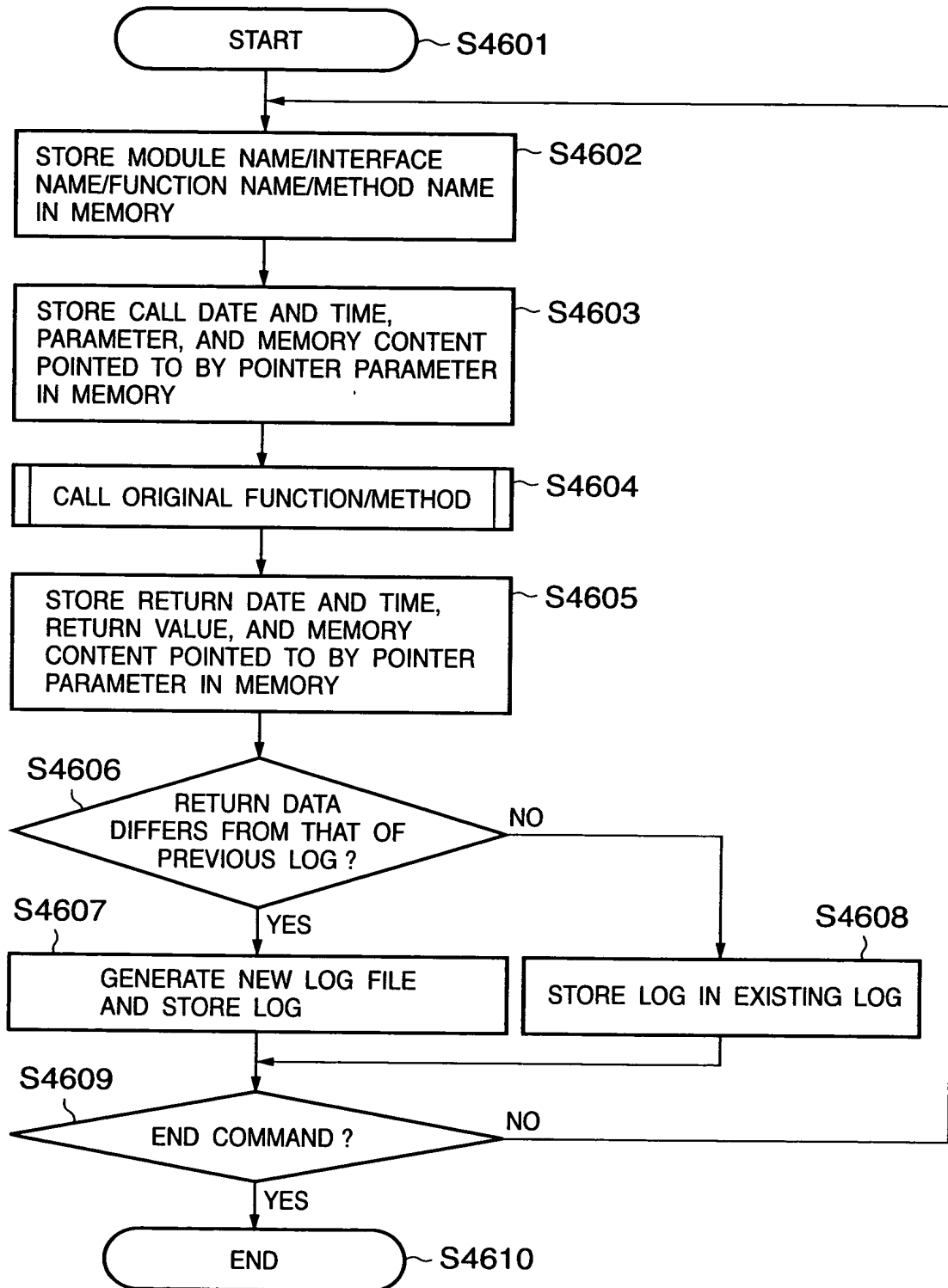
FIG. 45

FIG. 46

S4701

START

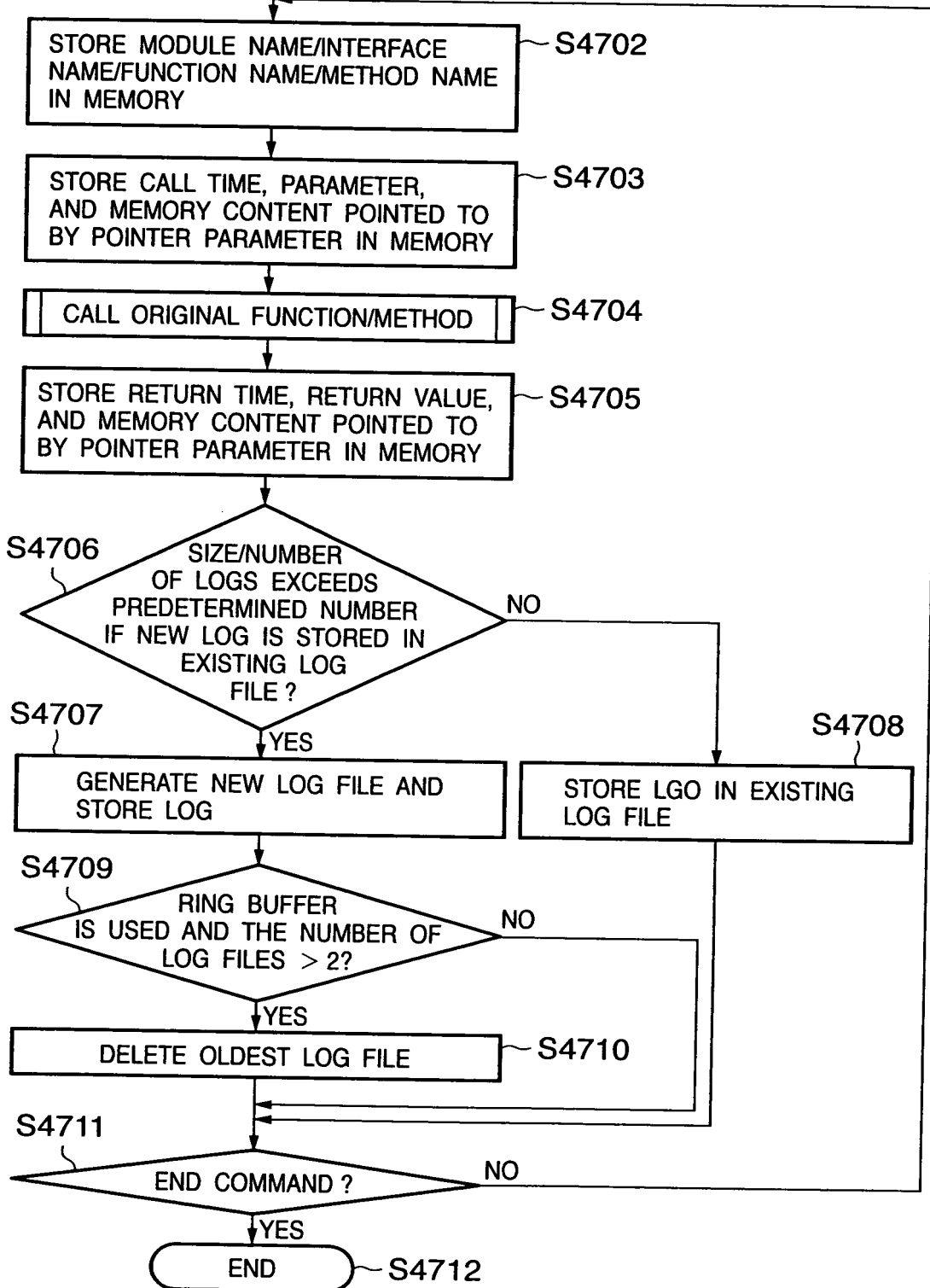
FIG. 47

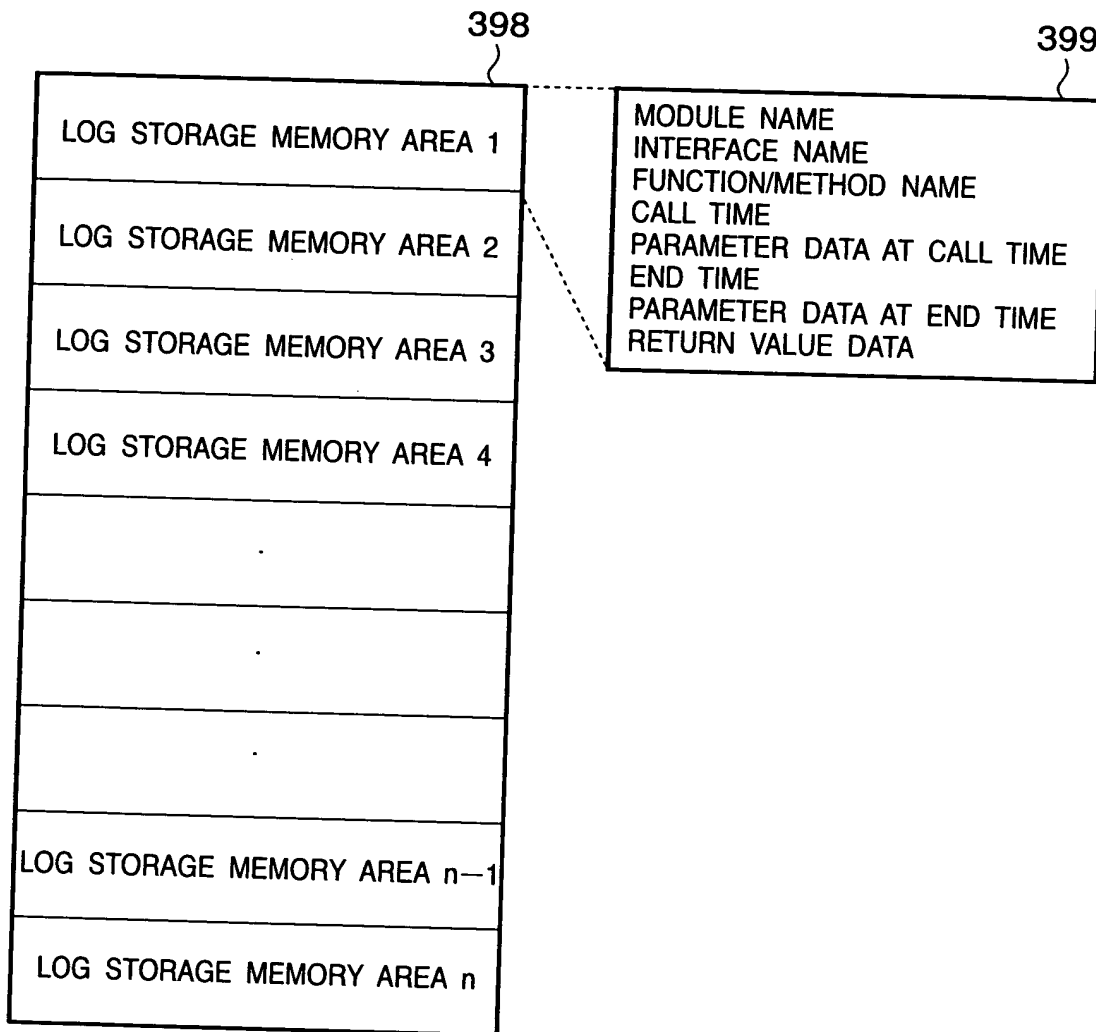
FIG. 48

FIG. 49